Nr. 5/86 Mai
3. Jahrgang -DM 5,50 / ÖS 46 / SRF 5,50

# COMMODORE WELT



Das unabhängige Commodore-Magazin

**AGRESSIV:** Commodore greift an

**UMFASSEND:** Neue Basic-Befehle für den VC 20

**GETESTET:** Echte Software für 128 PC

**PROFIHAFT:** 60 Zeichen am 64 er-Bildschirm

**INTENSIV:** CP/M-Das unbekannte Wesen

Tips, Tricks & Kaufberatung




Nr. 4/86/April

Das erste deutsche MSX-Magazin

# MSX: DER STANDARD FÜR IHREN PRIVATEN PC

**IM TEST:** Yamaha CX 5

**SERVICE:** Welcher Drucker für MSX?

**IM TEST:** Text-verarbeitung

**SERVICE:** Basic-Befehle

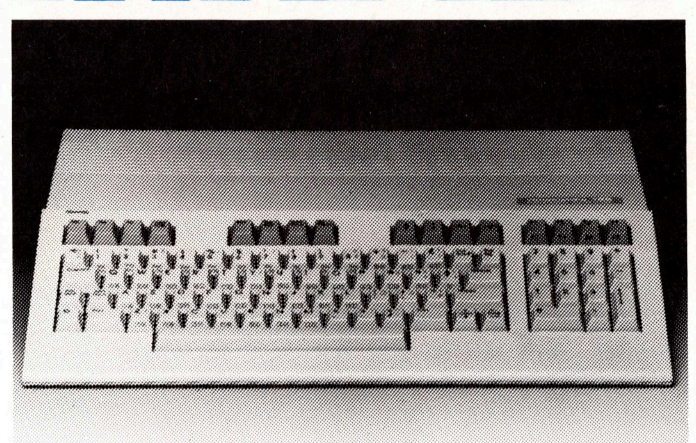
**IM TEST:** Joy-sticks

**SERVICE:** Über 20 Seiten MSX-Listings

# JETZT AN IHREM KIOSK

# INHALT

COMMODORE WELT
Nr. 5/86



## UNSER ☏ TELEFONSERVICE ☏

**Alle Experten der COMMODORE-WELT stehen unseren Lesern jeden Mittwoch zwischen 16.30 und 20.00 Uhr zur Beantwortung aller Fragen zur Verfügung unter der Telefonnummer 089/129 80 13. Ebenso der Abo- und Kassettenservice. Einfach anrufen 089/129 80 3/4!**



## IMPRESSUM


COMMODORE-WELT
erscheint monatlich in der AKTUELL-Gruppe Werner E. Seibt

VERANTWORTLICH FÜR DEN INHALT: Anton Kult/Werner E. Seibt

REDAKTION UND STÄNDIGE MITARBEITER: Senator-Presse-Service: Brigitte Gerl, Helmut Gerl, Rosemarie Huber, Dipl.-Ing. Rainer Krampe, Torsten Seibt, Georg Ziemann

ANZEIGENLEITUNG: Bruno Redase

ANSCHRIFT FÜR ALLE VERANTWORTLICHEN:
Postfach 1107, 8044 Unterschleißheim
Tel.: 089/129 80 14
Es gilt Preisliste Nr. 6 vom 1.1.86
Media-Unterlagen bitte anfordern

GRAFISCHE GESTALTUNG:
AK-Design, SPS

VERTRIEB: Verlagsunion Wiesbaden





Druck: Mayer und Söhne.
Erstverkaufstag: 25. April 1986

# GAME-KILLER: ALLE MACHT DEM SPIELER!

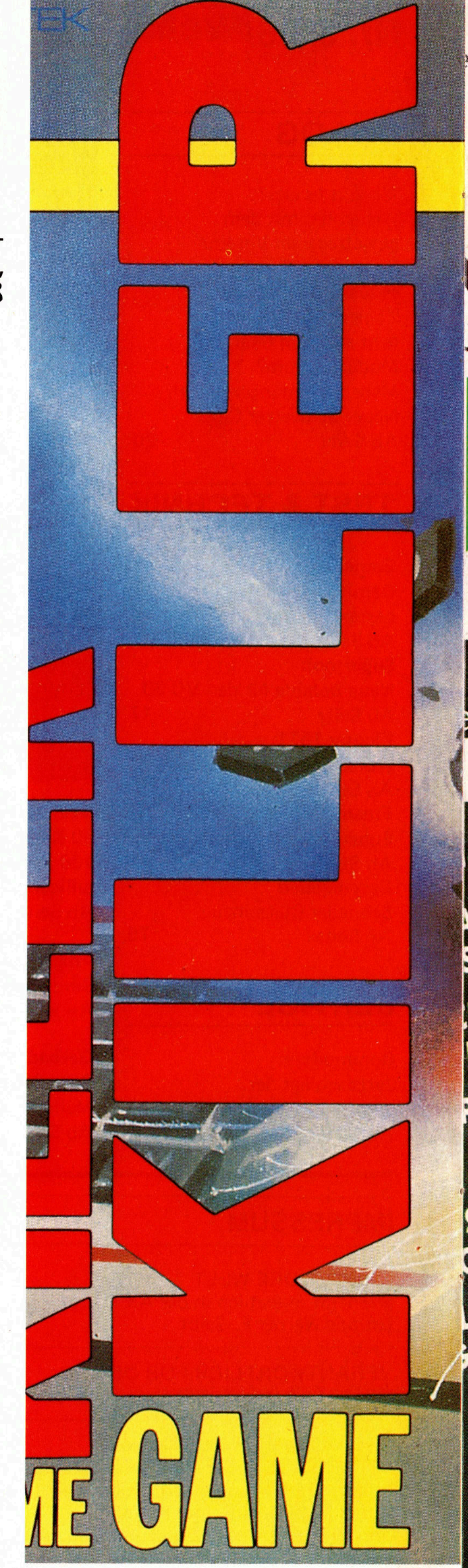
**Mastertronic hat wieder zugeschlagen. Der neueste Gag des Softwarehauses ist ein Modul für den 64er, welches, in den Userport gesteckt, ein Spiel ohne Grenzen verspricht. Dieses Modul soll dafür sorgen, daß bei eingeladenen Spielen die jeweilige Spielfigur „unsterblich" wird.**

Was bisher in einigen Spielen nur über entsprechende Befehlsroutinen möglich war, macht nun das Modul „Game-Killer" von Robtek im Vertrieb von Mastertronic allgemeingültig: Es wird in den User-Port des C 64 gesteckt und harrt dort der Aufgaben, die ihm bevorstehen. Nach Einschalten des Computers erscheint zuerst einmal das Titelbild des Game-Killers, untermalt von Computerklängen des Mastertronic-Spezialisten Rob Hubbard, welcher schon in diversen Mastertronic-Spielen bewiesen hat, daß sein Name mit höchster akustischer Qualität gleichzusetzen ist. Wer dann auf die Space-Taste drückt, erlebt schon gleich das erste Erstaunen: Unschuldig meldet sich das C 64 Titelbild, wie es eigentlich gleich nach demEinschalten erscheinen sollte.

## Kleiner roter Knopf

Nun kann, egal von welchem Speichermedium, das eigentliche Spiel in gewohnter Art und Weise eingeladen werden.
Der Spielverlauf wird solange nicht beeinflußt, bis ein kleiner roter Knopf auf der Modulseite gedrückt wird: Damit ist Game-Killer aktiviert und nun dürfte eigentlich dem Spieler nichts mehr „passieren". Hier allerdings mußte festgestellt werden, daß das Robtek-Modul nicht ganz den Versprechungen der beigelegten deutschen Beschreibung entsprach. Es gibt (laut Beschreibung) drei Möglichkeiten, um Game-Killer zu aktivieren:
1. Den Knopf an der Modulseite drücken: Alle Spritekollisionen werden damit ausgeschaltet.
2. Knopf drücken und gleichzeitig Joystick nach oben: Lediglich die Spritekollisionen des Gegners sind ausgeschaltet.
3. Knopf drücken und gleichzeitig Joystick nach unten: Die Hintergrund-Spritekollisionen sind desaktiviert.
4. Knopf und gleichzeitig Feuerknopf drücken: Game-Killer ist ausgeschaltet.
Zu den einzelnen Punkten:
Das allgemeine Aktivieren des Moduls mittels des seitlichen Knopfes geschieht problemlos, die versprochene Wirkung tritt ein. Will der spielbesessene Anwender nun aber die zweite Möglichkeit benutzen, wie es die Beschreibung verspricht, so muß er feststellen, daß sich die Wirkung in keinster Weise von der ersten Schaltvariante unterscheidet. Davon, daß der Gegner zwar getroffen wird, der Spieler aber unverletzt bleibt, ist leider nichts zu bemerken, denn beide erfreuen sich bester Gesundheit, egal wer wen beschießt. Die Geschoße jedenfalls durchfliegen Gegner wie Spieler, als ob beide nicht vorhanden wären. Ebenso verhält es sich mit Variante 3, auch hier ist kein Unterschied zur ersten Stufe zu bemerken. Zu Punkt 4 wäre zu sagen, daß es im Testverlauf kein einziges Mal mißlang, Game-Killer wärend eines Spieles auszuschalten.

Und nun zur Praxis: Wir versuchten Gamekiller an den verschiedensten Spielen, welche bei einem Spiele-Freak wohl zu finden sein werden. Hier wurden die unterschiedlichsten Reaktionen im Spielverlauf erzeugt, Langeweile kam mit Sicherheit nicht auf. In „Impossible Mission" beispielsweise funktionierte das Modul erst, nachdem der Held einen Raum betrat. Aktivierten wir das Modul bereits, wenn die Spielfigur noch auf einem Gang stand, so stürzte diese einfach durch sämtliche Böden hindurch und ward nie mehr gesehen. Bei be-






stimmten Schießspielen konnten einem die gegnerischen Kugeln zwar nichts mehr anhaben, die eigenen Geschosse allerdings kratzten den „Feind" genau so wenig. Natürlich ist es auch noch möglich, daß Game-Killer bei einem bestimmten Spiel überhaupt nichts hilft, auch dies erlebten wir während des Testverlaufs beispielsweise bei „Fast Fire". Nun gibt es aber Spiele, die (um sie zu beenden) auf Sprite-Kollisionen angewiesen sind, beispielsweise Grazy Kong, wo zum Erreichen des nächsten Levels eine Figur gerettet werden muß, und dies wird eben durch eine Spritekollision, sprich Berührung, erreicht. Daher ist es erforderlich, mitten im Spiel auf die oben beschriebene Weise Gamekiller auszuschalten, um ins nächste Level zu gelangen, dies geschieht allerdings, wie gesagt, problemlos. Mit der Allgemeinverträglichkeit des Moduls gab es ebenfalls keine Probleme, jedwede Software vom Spiel über Textverarbeitungsprogramme bis hin zum Simons Basic blieben von der Anwesenheit Game-Killers im Modulschacht ungestört und funktionierten einwandfrei. So muß also festgestellt werden, daß dieses Modul trotz der etwas zu hoch gegriffenen euphorischen Begleittexte doch eine ansehnliche Wirkung erzeugt.

## Wer will nicht den besten Freund schlagen?

Abgesehen von der Tatsache, daß einige Spiele weiterhin resistent gegen jede Art von äußerer Beeinflußung des Spielergebnisses sind, kann Game-Killer dem Kaufinteressierten trotzdem empfohlen werden, da der Verkaufspreis von 69,–– DM sicherlich nicht zu hoch gegriffen ist. Eine Auskunft bleibt die Beschreibung allerdings schuldig: Wozu das Modul eigentlich gut sein soll?

Nützlich erscheint es höchstens für die Verkäufer von Computerspielen, welche nun endlich vorführen können, was in ihren Produkten alles erreichbar ist. Oder aber für jene Spiele-Freaks, die endlich einmal den besten Freund schlagen wollen, welcher immer ein paar Punkte mehr erreicht. *T. Seibt*

# COMMODORE GREIFT AN UND SCHLÄGT ZURÜCK

**Wir sind – und wir bleiben – Nummer 1. Sagte Commodore. Und will jetzt die anbrandende Konkurrenz das Fürchten lehren. Unter anderem mit einem neuen C64.**

„1986 haben wir die Weichen für den künftigen Erfolg gestell". Mit diesen Worten trat Harald Speyer, Europa-Boß des amerikanischen Computerherstellers, auf einer Pressekonferenz während der Hannover-Messe CeBIT dem Krisengerede um den C 64 im Markt hatten". Doch das habe sich inzwischen geändert. „Aller Wahrscheinlichkeit nach werden wir im Juni auch noch den in Braunschweig entwickelten PC auch in den Staaten anbieten".

Insgesamt hat Commodore laut tet Speyer „von unserem Mediencomputer einen noch größeren Erfolg als in den USA (50 000 mal verkauft), vor allem im gehobenen Heim- und im professionellen Bereich".

Inzwischen gab Nigel Shepher, General Manager von Commodore North America, eine starke Preisreduzierung für den Amiga 1000 bekannt. Um alle noch zögernden Interessenten in den USA zu ködern, wurde der Amiga um 500,– Dollar auf 795,– Dollar gesenkt. Legt man einen Wechselkurs von 2,60 DM zu Grunde, kostet der Amiga nun nur noch 3354,– DM. Allerdings gilt dieses Angebot nur für ein Komplettsystem mit Farbmonitor 1080 (495,– Dollar). Ob und wann uns dieser Preisnachlaß auch in




**Immer noch der meistverkaufte Home-Computer aller Zeiten: C64. Allein zwei Millionen davon konnten 1985 abgesetzt werden.**

das Unternehmen entgegen. Das US-Management sei komplett ausgewechselt, die Probleme mit den Banken seien ausgestanden und zwei marode Betriebe geschlossen worden.

Die Schwierigkeiten in den USA seien entstanden, „weil wir dort nur Speyer seit 1977 über 10 Millionen Computer verkauft, davon eine Million PC, sechs Mio. C 64 – 2 Mio. allein 1985 – und inzwischen bereits fast eine halbe Million C 128. Speyer: „Die erfolgreichste Markteinführung aller Zeiten".

Vor allem in Deutschland erwar- Deutschland erreicht, war bis Redaktionsschluß noch nicht in Erfahrung zu bringen. Auf jeden Fall dürfte die Konkurrenz in dieser Preis- und Leistungsklasse damit schwer ins Abseits gedrängt werden.

Ein MS-DOS-Simulator (softwaremäßig) und eine Btx-Box wurden

für die nahe Zukunft (was immer das heißen mag) angekündigt. Absatzerwartungen für Commodores neues Prachtstück: „5000 bis Juni".
Auch bei der IBM-kompatiblen Serie PC 10/20 hat sich bei Commodore einiges getan.

Schon rein äußerlich haben sich die PC's zu ihrem Vorteil verändert. Tastatur und CPU-Gehäuse wirken nun durch ihre abgerundeten Kanten futuristischer und gliedern sich damit rein optisch besser in die moderne Bürolandschaft ein.

Im Innenleben wurden sie nun gleich auf der Hauptplatine (es gibt nur noch eine) auf die doppelte RAM-Speicherkapazität (512KB) aufgebohrt. Damit verbleibt dem Anwender noch eine stille Reserve von 128KB, bis er die Höchstgrenze erreicht hat. Ärgerlich ist nur, daß diese 128 KB, werden sie unbedingt benötigt, dann einen Steckplatz belegen.

512KB sind aber auf jeden Fall ausreichend genug, um das Gros der Programme für die MS-DOS-Maschinen starten zu können.

Serienmäßig hat Commodore jetzt auch in alle Geräte Farbgrafikkarten eingebaut.

Nicht unerwartet wurde der Preis, trotz mehr an Leistung, für alle Geräte weiter gesenkt. Kostete der PC 10 bei seiner Einführung im vergangenen Jahr noch fast 6000,– DM, steht er nun bei den Händlern nur noch mit 3995,– DM incl. Mehrwertsteuer in der Liste.

Dem PC 20 hat Commodore nicht nur einen doppelten RAM-Speicher spendiert, sondern liefert ihn auch gleich mit einer neuen 20 Megabyte-Festplatte aus, zu einem Preis der anfangs noch für seinen kleinen Bruder verlangt wurde. Es ist eigentlich zu erwarten, daß die Fehlerhäufigkeit im Festplattenbereich durch den Einbau eines neuen Modells zumindest stark eingeschränkt, wenn nicht sogar völlig ausgeschlossen ist.

Neben der geänderten Form ist die neue Serie an der römischen Zwei hinter der Gerätebezeichnung (PC 10-II; PC 20-II) zu erkennen. Commodore sieht diese erweiterte und preisgünstigere Serie als klare Antwort auf die Diskontwelle aus Taiwan und Korea an. Vollkommen überraschend präsentierte Commodore auf der Ce-BIT in Hannover



Der Commodore-Hoffnungsträger Amiga. Bald für unter 35000 DM erhältlich?

## Da kriegen selbst die Taiwanesen feuchte Augen: Ein PC AT für unter 10 000 Mark

eine weitere kleine Sensation, einen IBM-AT kompatiblen Rechner für unter 10 000,– DM. Der Büromaschinenhersteller nannte ihn schlicht und einfach „Commodore AT". Nicht der Name ist wichtig, sondern die Leistung und in dieser Beziehung hat Big Blue einen weiteren harten Brocken zu schlukken, obwohl der Computerriese mittlerweile auch ganz schön an der Preisspirale gedreht hat. Das neue Flaggschiff besitzt selbstverständlich gleich serienmäßig einen 640KM-Arbeitsspeicher, die bereits bekannte Grafikkarte, die ja auch eigentlich für dieses Gerät entwickelt wurde, eine 3,5" Festplatte mit 20 MB, fünf freie Steckplätze, 14" Monitor und paralelles und serielles Interface. Optional kann Commodores neueste Errungenschafft noch um eine 40MB-Festplatte erweitert werden.

## Im Sommer kommt ein neuer 64er. (Zumindest ein neues Gehäuse.)

Um den 64er ist es im Augenblick etwas still gewordne. Angesichts der Macht des Amigas mußte er sich am Commodorestand in ein dunkles Eckchen zurückziehen. Die Konkurrenz im Homecomputersektor frohlockte schon, daß die Produktion und die Beliebtheit des meistverkauften Homecomputers der Welt ihrem Ende entgegen geht. Zwar noch unbestätigt, aber es wird noch in diesem Sommer eine neue und verbesserte Version die Kaufhäuser erobern. Die Gerüchteküche spricht hierbei von einer Version mit doppeltem Arbeitsspeicher (128KB), besserem Betriebssystem und einem anderen Basicdialekt.

Inwieweit sich die Gerüchte bestätigen,werden bleibt vorerst noch abzuwarten.

# AUF DER SCHNELLSTEN RENNSTRECKE DER WELT




Steuern Sie mit dem Joystick in Port 1 Ihren Rennwagen über die schnellste Rennstrecke der Welt.
Doch so leicht wird Ihnen

**Rasantes Listing**

das bestimmt nicht fallen, denn erst die Übung macht den Rennfahrer. Und sollte Ihnen Ihr Rennen zu einfach erscheinen, so können Sie mit insgesamt 729 (!) Möglichkeiten den Schwierigkeitsgrad von der Geschwindigkeit bis zur Fahrbahnbreite selbst gestalten.
Langweilig dürfte es Ihnen also nicht so schnell werden, wie Ihr Auto über die Strecke fegt.

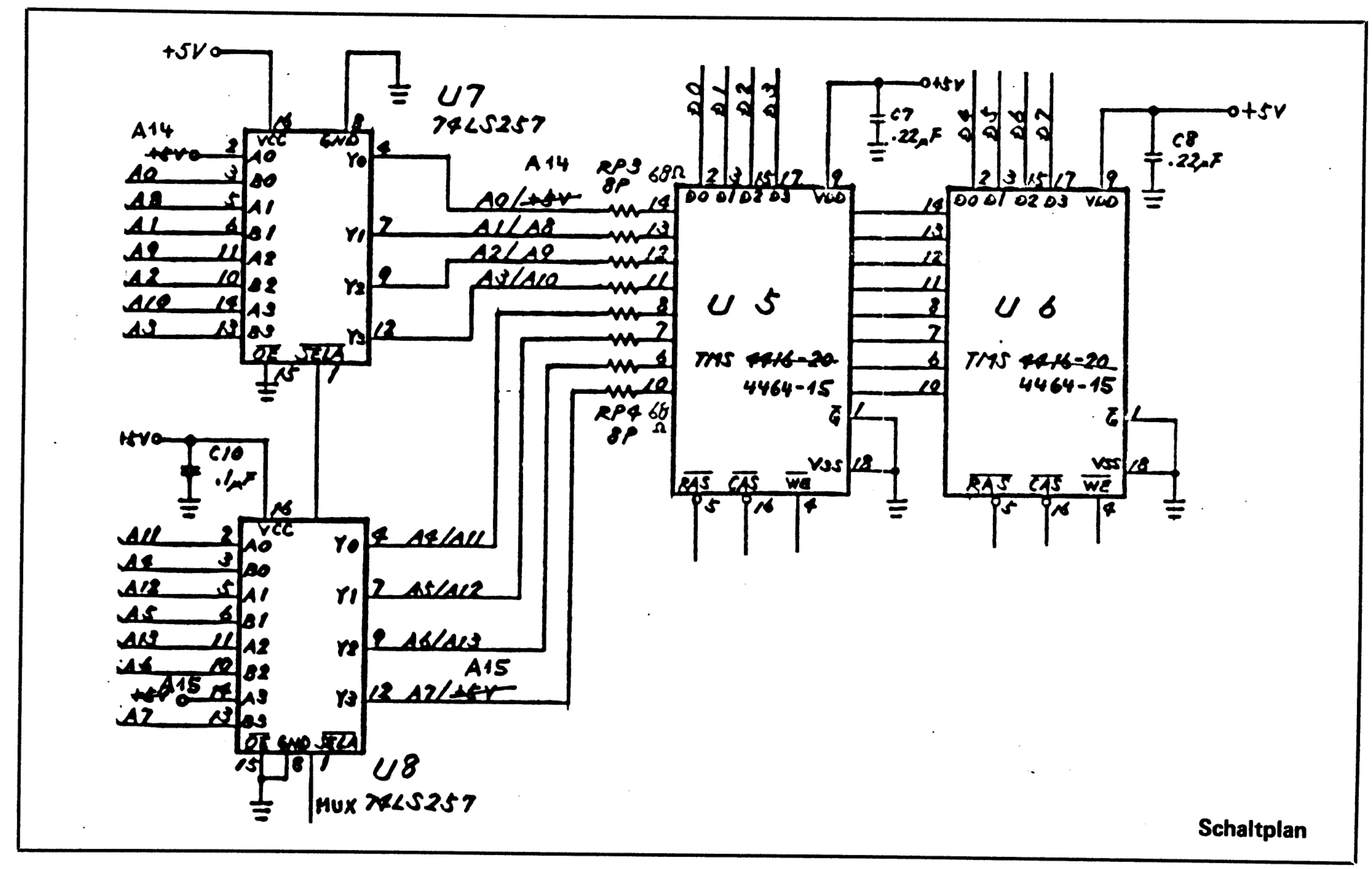
# Maserati — 1 —

```
10 rem maserati =================64
20 rem (p) commodore welt          =
30 rem =============================
40 rem (c) by                      =
50 rem klaus dillinger             =
60 rem                             =
70 rem                             =
80 rem c-64 + 1530/1541            =
90 rem =============================

100 cl$=chr$(147):wh$=chr$(5):c4$=c
hr$(17):cy$=chr$(159)
110 p=54272:v=53248:pokev+32,0:poke
v+33,0
120 printcl$c4$c4$c4$tab(8)wh$"gesc
hwindigkeit (1-9)"
130 geta$:ifa$=""then130
140 ifasc(a$)<49orasc(a$)>57then130
150 printc4$c4$c4$tab(8)"fahrbahnbr
eite (1-9)"
160 getb$:ifb$=""then160
170 ifasc(b$)<49orasc(b$)>57then160
180 printc4$c4$c4$tab(8)"schwierigk
eitsgrad(1-9)"
190 getc$:ifc$=""then190
200 ifasc(c$)<49orasc(c$)>57then190
210 printcl$"wait a moment."
220 ifs=0thenfort=0to62:reada:poke7
04+t,a:s=s+a:next:poke2040,11
230 ifs<>9558thenprint"fehler in sp
rite-datas":end
240 ifl=0thenfort=49152to49395:read
a:l=l+a:poket,a:next
250 ifl<>32178thenprint"fehler in m
aschinenprogramm-datas":end
260 a=10-val(a$):b=val(b$):c=val(c$
):d=6+b:poke181,d:poke182,a
270 poke183,c:pokev,172:pokev+1,210
280 printcy$cl$;:fort=1to24:printta
b(20-d/2)"*"tab(20+d/2)"*":next
290 pokev+21,1:poke185,2:poke186,1:
poke187,peek(183):poke188,20-d/2
300 x=peek(v+31):poke191,10:pokep+2
4,15:pokep+5,100:pokep+6,100:pokep+
4,33
310 pokev+39,7:poke140,10:sys49205
320 pokep+4,0:pokep+5,0:pokep+6,0
330 pokep+8,10:pokep+13,218:pokep+1
1,129
340 pokep+24,0:pokep+11,128:fort=1t
o15:pokep+24,t:next
350 pokev+21,0:printwh$cl$c4$c4$c4$
tab(5)"noch ein versuch(j/n)"
360 getr$:ifr$<>"j"andr$<>"n"then36
0
370 ifr$="j"thengoto120
380 rem datas
390 data1,129,128,3,255,192,27,219,
216,27,219,216,31,195,248,31,255,24
8,27
400 data255,216,27,255,216,3,255,19
2,3,255,192,7,153,224,119,189,238,1
19
410 data165,238,119,66,238,126,90,1
26,126,126,126,119,60,238,119,129,2
38
420 data15,255,240,31,227,248,63,19
3,252
430 rem datas maschinenprogramm
440 data162,24,160,7,132,140,132,14
2,160,152,132,139,160,192,132,141,1
60
450 data0,177,139,145,141,200,192,3
9,208,247,165,139,56,233,40,176,2,1
98
460 data140,133,139,165,141,56,233,
40,133,141,176,2,198,142,202,208,22
0,96
470 data32,231,192,32,82,192,32,97,
192,198,191,208,243,169,5,133,191,3
2,0
480 data192,32,141,192,173,31,208,2
40,228,96,160,0,162,0,232,224,40,20
8,251
490 data200,196,182,208,244,96,198,
185,240,1,96,169,1,133,185,173,1,22
0,41
500 data4,208,10,206,0,208,208,5,16
9,0,141,16,208,173,1,220,41,8,208,1
0,238
510 data0,208,208,5,169,1,141,16,20
8,96,160,39,169,32,153,0,4,136,208,
250
520 data198,187,208,25,165,186,201,
1,208,7,169,255,133,186,76,172,192,
169
530 data1,133,186,165,162,41,15,101
,183,133,187,165,186,201,1,208,2,23
0,188
540 data165,186,201,255,208,2,198,1
88,165,188,169,4,133,189,160,0,169,
102
550 data145,188,165,181,168,169,102
,145,188,165,188,201,5,48,5,201,23,
176
560 data1,96,169,1,133,187,96,230,1
39,165,139,41,7,105,3,141,1,212,96,
0
570 rem masarati =============64
580 rem 38911 bytes memory     ===
590 rem 02807 bytes program    ===
600 rem 00126 bytes variables  ===
610 rem 00000 bytes arrays     ===
620 rem 00025 bytes strings    ===
630 rem 35953 bytes free (0)   ===
640 rem ========================
```

# KEIN APRILSCHERZ! 60 KB FÜR IHREN C16/116!

**Mit dieser Schaltung, sie kostet – einschließlich Fremdarbeit – ganze 100 Mark, haben Sie im C 16/116 plötzlich über 60 Kbytes im RAM! Wirklich. Sie können es aber auch selbst machen.**




Schaltplan

60671 BYTES FREE! Wenn sich Ihr Computer mit diesem Text meldet, ist aus Ihrem C 16 ein C 164 bzw. aus Ihrem C 116 ein C 1164 mit insgesamt 64 K RAM geworden, das heißt: mehr freier Speicherplatz als der legendäre C 64!!!

Sie fragen sich, wie das möglich ist? Ich zeige es Ihnen, damit Sie die Änderung selbst durchführen können. Doch Achtung! Bei Eingriffen in das Gerät erlischt die Garantiepflicht des Herstellers. Dies ist jedoch nur dann von Bedeutung, wenn Ihr Gerät noch innerhalb der Grantiezeit von 6 Monaten nach dem Kaufdatum liegt. Doch erst ein kurzer Überblick über den Schreib- Lesespeicher Ihres Computers.

Im Schaltbild sind die zwei Multiplexer (Umschalter) SN74KS257 (U7+U8) sowie die zwei DRAMs (dynamische RAMs) d.h. Schreib-, Lesespeicher TMS 4416-20 (U5 + U6) mit den 8 Vorwiderständen RP3 + 4 zu sehen. Die DRAMs benötigen eine Stromversorgung an den Pins 9 + 18 sowie Refreshsignale (zum Auffrischen der Speicherinhalte) an den Eingängen RAS und CAS Pin 15 + 16. Der Gateeingang G ist fest an Masse und damit offen, die Steuerung des DRAMs übernimmt der Write-Enable-Eingang WE an Pin 4. Der TMS 4416-20 ist aus der 4000er Serie, die zweite Ziffer von links gibt die Bitbreite (hier 4) an den Anschlüssen 2,3,15 und 17 an, die 3. + 4. Ziffer von links die Speichertiefe von 16 K an. Die unteren 4 Bit des vom Prozessor 8501 benutzten 8 Bitwortes DO-D3 (D= Da-

tenbit) liegen an U5, die oberen 4 Bit D4-D7 liegen an U6. Die Zahl rechts vom Bindestrich gibt mit 10 multipliziert die Zugriffszeit (Geschwindigkeit) des DAMs in ns (Nonosekunden) an. Setzen wir den Baustein des RAM-Bereiches von 16 auf 64 K in der 3. + 4. Ziffer und eine Vergrößerung der Geschwindigkeit von 25 % von 200 auf 150 ns. Bereits für 16 K d.h. 16384 Speicherstellen, benötigen wir mindestens 14 Adressleitungen (214), mehr als Adressanschlüsse vorhanden sind. Um dieses Problem zu lösen, sind die Multiplexer U7 + U8 vorgeschaltet, die mit dem Multiplexsignal MUX an Pin 1 zwischen den unteren Adressbits (A0-A7) und den oberen 8 Adressbits (A8-A15) umschalten. Da wir für 64 K d.h. 65536 Speicherstellen 16 Adressleitungen (216) benötigen, müssen die bisher unbenutzten MUX-Eingänge U7/2 und U8/14 von +5V getrennt und an die oberen beiden Adressbits A14 und A15 gelegt werden.

Nun zur Umbauanleitung:

Nachdem die Bodenschrauben gelöst sind, läßt sich der Boden nach hinten verschieben und dann nach vorne herausklappen. Die Stecker für das Keyboard und das ON LED werden abgezogen. Das Abschirmblech wird durch Herausdrehen der Schrauben gelöst und abgenommen. Nun kann man die Computerplatine herausnehmen und nach einseitigem Aufbiegen der Blechlaschen auch die untere Abdeckung entfernen. Die DRAM U5 + U6 werden nun am besten mit einem IC-Setzer geklammert, mit einem DIL-Entlöter erwärmt und vorsichtig herausgezogen. Achtung, einige Beinchen können umgebogen sein und haken. Sind diese Spezialwerkzeuge nicht vorhanden, können die Beinchen der ICs auch an den IC-Körpern mit einem sehr spitzen Elektronik-Schneider abgeschnitten, die IC entfernt und die einzelnen Beinchen mit einer Pinzette und einem Lötkolben aus der Leiterplatte ausgelötet werden. Die Löcher sind mittels eines Lotsaugers oder Lotsauglitze von Zinnresten freizulegen. Jedoch wiederum Achtung! Die ICs sitzen in durchkontaktierten Löchern in einem Layout in Feinleitertechnik. Die Beinchen gehen schwer heraus, die dünnen Leiterbahnen reißen schnell ab. Dies wäre dann der Anfang vom Ende Ihres Computers. **Schicken Sie mir bitte nicht solche Krüppel!** Haben Sie bis hierher alles richtig gemacht, setzen Sie die neuen ICs TMS 4464-15 ein und verlöten Sie die Lötstellen. Zum Löten sollte ein Trenntrafolötkolben Verwendung finden, die ICs dürfen nicht überhitzt werden. Jetzt müssen die +5V-Verbindungen zu U7/2 und U8/14 aufgetrennt werden. Die Leiterbahn zu U8/14 läßt sich leicht auf der Unterseite auftrennen, die Leiterbahn zu U7/2 verläuft unter dem IC und es wäre unsinnig, hierfür das IC auszubauen. Man kneift mit einem spitzen Elektronik-Schneider einfach das Beinchen kurz über der Lötstelle ab und biegt es hoch. Jetzt braucht man nur noch die Verbindungen A14 an U1/37 zu A0 an U7/2 sowie A15 an U1/36 zu A3 an U8/14 mit einem dünnen Draht, am besten von Beinchen zu Beinchen auf der LPL -Oberseite, hergestellt werden. Der so umgebaute Computer sollte, in umgekehrter Reihenfolge zusammengebaut, jetzt die eingangs erwähnte Meldung „60671 Bytes FREE" zeigen. Tut er dies — meinen Glückwunsch!!!

Es sei jedoch darauf hingewiesen, daß auch versierte Fachleute beim Auslöten von ICs aus durchkontaktierten Löchern beim Layout in Feinleitertechnik ihre Probleme haben. Ich würde Ihnen raten, fall Sie sich nicht absolut sicher sind den Umbau problemlos durchzuführen, mir Ihren Computer in guter Verpackung, ohne Zubehör, zum Umbau zuzusenden. Ich baue Ihnen Ihren Computer innerhalb einer Woche für 97,80 DM incl. aller Kosten für Material, Einbau, MWSt. und Rückporto um. Senden Sie mir Ihren Computer mit 100,00 DM per Vorkasse (die restlichen 2,20 DM muß ich der Post geben, wenn Ihr Paket ankommt) in bar, per Scheck oder auf eines meiner Konten bei der Kreissparkasse Segeberg, BLZ 23051030 Konto 557064 bzw. Postscheckamt Hamburg, BLZ 20010020, Konto 57494-205, unter Angabe von Namen und Adresse. Meine Adresse lautet: Elektronik-Technik,Tannenweg 9, 2351 Trappenkamp.

Viel Spaß Euer *upe*

# HÄTTEN SIE ES GEWUSST?

Info Plus/4: C-128 Color in BASIC 3.5 : BASIC 7.0

In der BASIC-Version 3.5
COLOR: Farbquelle, Farbcode, Farbhelligkeit
Farbquelle:
0 Bildschirm Hintergrund 40 Zeichen, 1 Bildschirm Vordergrund, 2 Grafischer Mehrfarbenmodus 1, 3 Grafischer Mehrfarbenmodus 2, 4 Bildschirm Rand, 5 nicht belegt, 6 nicht belegt.
Farbcode:
1 Schwarz, 2 Weiß, 3 Rot, 4 Cyan, 5 Purpur, 6 Dunkelgrün, 7 Blau, 8 Gelb, 9 Orange, 10 Braun, 11 Gelbgrün, 12 Rosa, 13 Balugrün, 14 Hellblau, 15 Dunkelblau, 16 Hellgrün.

In der BASIC-Version 7.0:
COLOR: Farbquelle, Farbcode.
Farbquelle:
0 Bildschirm Hintergrund 40 Zeichen, 1 Grafischer Vordergrund, 2 Grafischer Mehrfarbenmodus 1, 3 Grafischer Mehrfarbenmodus 2, 4 Bildschirm Rand, 5 Textfarbe, 6 Bildschirm Hintergrund 80 Zeichen.
Farbcode:
1 Schwarz, 2 Weiß, 3 Rot, 4 Cyan 1/40Z 5 Purpur 1/40Z, 6 Dunkelgrün, 7 Blau, 8 Gelb, 9 Hellbraun (Pupur 2/80Z, 10 Braun, 11 Rosa, 12 Dunkelgrau (Cyan 2/80Z, 30 Mittelgrau, 14 Hellgrün, 15 Hellblau, 16 Hellgrau.
Helligkeit/Luminance
0 = dunkel, 7 = hell.
Jede Farbe ist auch noch in ihrer Helligkeit /Luminance (außer Schwarz) veränderbar.

# IF YOU FREAK THEN GOTO DOWN

**Sind Sie ein Computerfreak? Ein richtiger? Sie wissen es nicht? Kreuzen Sie doch einfach unseren Fragebogen an, wir geben Ihnen dann die Antwort, haften aber nicht für Folgeschäden!**

Ein Test für alle, die ausser PIEP, ZONG und RUMS auch noch „Einmal die COMMODORE WELT bitte“ sagen können.
Gebrauchsanleitung:
Take Stift.
For Frage = 1 to alle Fragen
Beantworte Frage durch Ankreuzen von vorgegebener Antwort
Next Frage
For Punkte = 1 to alle Punkte
Gesamtpunktzahl = Gesamtpunktzahl + jeweilige Punkte
Next Punkte
Look on Seite 63
If Punktzahl = 1. Spalte Then Read 1. Spalte; goto ende
If Punktzahl = 2. Spalte Then Read 2. Spalte; goto ende
Read 3. Spalte
ende

Übersetzt und somit auch für „normale“ Bürger verständlich lautet das ganze folgendermaßen:
Gleich nach dieser Gebrauchsanleitung werden Sie mehrere Fragen finden, die Sie nach bestem Wissen und Gewissen beantworten sollten.
Um Ihnen Ihre Antwort zu erleichtern, sind Ihnen schon ein paar Antworten vorgegeben; bitte kreuzen Sie die Antwortmöglichkeit an, die bei der jeweiligen Frage auf Sie am ehesten zutrifft. Selbstverständlich sollten Sie sich vorher einen FUNKTIONSTÜCHTIGEN Stift geholt haben, um Ihre jeweilige Antwort anzukreuzen. Am Ende des Testes zählen Sie die Punkte, die hinter jedem von Ihnen gemachten Kreuz stehen, zusammen und schauen auf Seite 63 nach, was für ein Ergebnis der Test für Sie bereithält.

Fragen:

1.)
Haben Sie während Ihrer Arbeitszeit fernab der heimischen Computeranlage des öfteren Ausbrüche von Angstzuständen, weil Sie befürchten, daß Ihrem Gerät etwas zugestossen sein könnte?
a) Ja, immer 5 P.
b) selten 1 P.
c) nie 0 P.

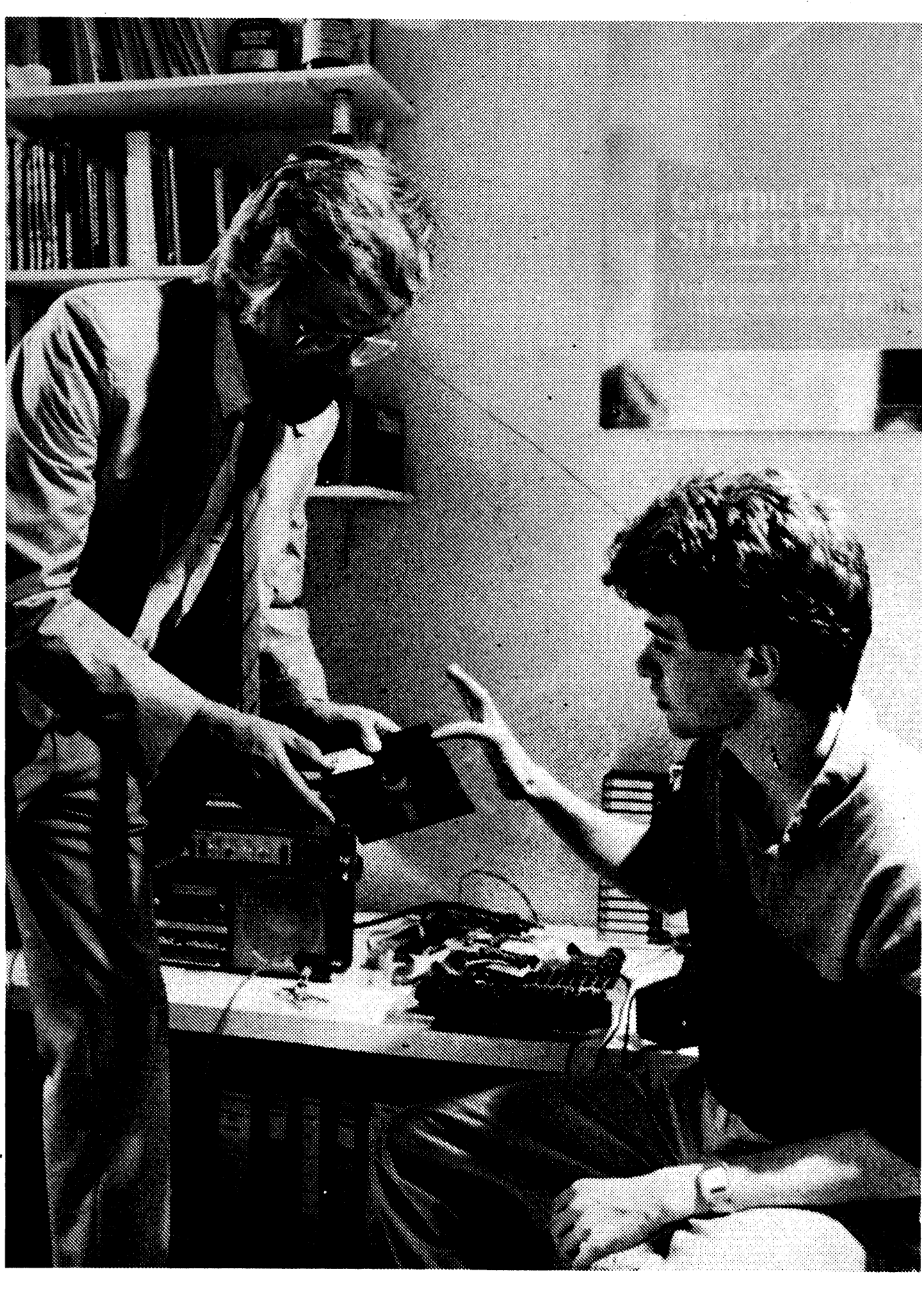
2.)
Ihr Haus brennt lichterloh.
Frau, Kind und Computer sind von den Flammen eingeschlossen. Was tun Sie?

a) Ich stürme in das Haus und rette Frau und Kind 0 P.

b) Ich stürme in das Haus und rette meinen Computer 4 P.

c) Ich gehe in den nächsten Computerladen und suche mir einen noch größeren Computer aus, den ich mit dem Geld von der Versicherung bezahlen kann. 5 P.

3.)
Stromausfall!
In Ihrem Stadtviertel wird eine Nacht lang der Strom abgeschaltet. Nach wie vielen Stunden bekommen Sie „Computerentzugserscheinungen“?



a) Ich bleibe cool und warte ruhig bis zum nächsten Tag ab 0 P.

b) Spätestens nach zwei Stunden ohne Computer nach dem Feierabend bin ich mit den Nerven völlig runter 1 P.

c) Allein schon durch den Gedanken daran bin ich kreidebleich geworden und brauche erst mal eine Schachtel Zigaretten, bevor ich mein Kreuz machen kann 3 P.

d) Ich habe für solche Fälle vorgesorgt und brauche nur noch mein Notstromaggregat anzuschalten 5 P.

4.)
Ihr Computer wird von gewissenlosen Kriminellen entführt. Als Lösegeld fordern die Entführer den halben Neuwert des Computers. Was tun Sie?

a) Ich zahle zähneknirschend das Lösegeld 3 P.

b) Ich informiere die Polizei, die GSG 9, die Bundeswehr und die Nato-Streitkräfte über den vorliegenden Sachverhalt 4 P.

c) ich kaufe mir endlich einen größeren Computer 5 P.

5.)
Ihre Frau hat versucht, den Computer mit einem nassen Lappen sauber zu machen, was zu einer absoluten Vernichtung desselben geführt hat. Wie reagieren Sie?

a) Ich beherrsche mich und streiche meiner Frau nur das Geld für Privatausgaben in den nächsten 10 Jahren 0 P.

b) Ich stelle meine Frau wegen fahrlässiger Tötung vor Gericht und lasse mich von ihr scheiden 4 P.

c) Im Affekt hole ich meinen schweren Revolver aus meiner Armeezeit und strecke meine Alte damit nieder 5 P.

6.)
Dritter Weltkrieg!!!
Die Russen nahen und wollen sowohl Ihre Frau als auch Ihre EDV-Anlage entführen. Was machen Sie?

a) Ich rette mich mit meiner Familie OHNE Computer in unseren Atomschutzbunker 0 P.

b) Ich lade mein Lieblingsprogramm in den Computer ein, um die letzten Minuten meines Lebens auszukosten 2 P.

c) Ich schnappe mir meinen Computer und schieße mir den Weg zum Atomschutzbunker gnadenlos frei 3 P.

d) Ich habe diese Situation vorausgesehen und brauche nur noch mein Anti-Missile-Defenseprogramm einzuladen, das meine in mühevoller Heimarbeit hergestellten Abwehrgeschütze aktiviert. Den Rest erledigt mein Computer 5 P.

# GANZ PROFESSIONELL: 60 ZEICHEN AUF DEM MONITOR

**40 Zeichen pro Zeile sind nicht genug – hier sind 60 Zeichen softwaremäßig.**

Für viele Anwendungen sind die 40 Zeichen pro Zeile des C64 nicht genug. Deshalb gibt es 80-Zeichen-Karten, die hardwaremäßig 80 Zeichen auf dem Bildschirm darstellen. Das ist jedoch ziemlich teuer. Ferner gibt es noch 80 Zeichen softwaremäßig. Dieses Verfahren funktioniert so: Die Zeichen werden in eine HIRES-Graphik-Seite eingezeichnet. Dabei stehen jedoch nur 40*8=320 Punkte in der Horizontalen zur Verfügung und somit für jedes Zeichen nur 4 Bit. Als Zwischenraum wird davon noch eines gebraucht – da waren's nur noch drei. Versuchen Sie mal ein drei Bit breites M zu malen. Hier sieht man schon einen Nachteil dieser Methode. Der zweite ist der große Speicherplatzbedarf eines HIRES-Bildes von 8 KByte.

Für viele Zwecke wäre man jedoch auch schon mit 60 Zeichen in einer Zeile zufrieden. Einen Brief schreibt man zum Beispiel meist mit 60 Zeichen. Auch eine Anleitung für ein Programm kann mit 60 Zeichen je Zeile mehr Text auf einer Seite unterbringen.

Wie aber soll man 60 Zeichen auf eine Zeile bringen, ohne einen HIRES-Graphik-Bildschirm zu verwenden? Nun, man nimmt zwei Bildschirme und arbeitet mit einem besonderen Zeichensatz (4–5 Bit breit). Dies sei an einem Beispiel gezeigt:

```
1234567812345678
****__***__****_
*__*__*__*_*__*_
*__*__*__*_*__*_
****__***__****_
*__*__*__*_*__*_
*__*__*__*_*__*_
*__*__*__*_*__*_
*__*__***__*__*_
----------------
```

Tippen Sie das Programm doch einmal ab und ersetzen die Zeile 85 durch: 85 for i=i to b:poke s+i, (speek (i+53248+2*b) and 60)*4: next 86 fori=1 to b: poke s+i+b, (peek(i+53248+2*b) and 60) /2: next
Die Zeilen 290–740 können Sie dazu zunächst einmal weglassen.

Der C64 hat zwei Zeichensätze gespeichert (im ROM ab 53248) und zwar den Groß-/Graphik- und den Klein-/Groß-Zeichensatz. Zusammen mit den entsprechenden reversen Zeichen ergibt das 4 KByte. Der Bildschirmcode eines reversen Zeichens unterscheidet sich von einem normalen dadurch, daß das 7. hoch 7 = 128). Das Programm erzeugt nun einen normalen Zeichensatz mit 4 Bit breiten Zeichen (‚and 60'), welcher um 2 Bit nach links verschoben ist (‚*4') und anstelle des reversen Zeichensatzes das gleiche um 2 Bit nach rechts verschoben (‚/2'). Dadurch entsteht zwischen einem normalen ‚A' und einesm reversen ‚A' ein Zwischenraum von 7 Bit, genug Platz für ein 3. Zeichen (hier das ‚B'). Dieses wird in den zweiten Bildschirm gepoket und zwischen beiden Bildschirmen wird 50 mal pro Sekunde umgeschaltet, wobei der zweite Bildschirm um 6 Bit nach rechts verschoben ist.

## Zwei Bildschirme im Interlace-Verfahren

Ein normales Fernsehbild besteht aus zwei Halbbilder (1., 3., 5 ... Zeile bzw. 2., 4., 6 ... Zeile), die nacheinander abgebildet werden (Interlaceverfahren). Commodore arbeitet mit zwei gleichen Halbbildern, 50 mal je Sekunde. Das Maschinenprogramm (SYS51236) schaltet nun zwischen diesen beiden Halbbildern jeweils auf den anderen Bildschirm um, so daß jeder 25 mal pro Sekunde abgebildet wird. Leider ist das Programm anscheinend noch nicht schnell genug, so daß ein Flimmern zu sehen ist. Es wird noch daran gearbeitet, dieses Flimmern zu verhindern. Wenn einem der Leser ein besseres Umschaltprogramm gelingt, soll er schreiben.

## Das Programm kann noch mehr

Der Ablauf des Programms isr aus den REM-Zeilen zu ersehen. Hier noch ein paar Tips, wie Sie das Maschinenprogramm für andere Zwecke verwenden können:
SYS 51200 kopiert den Originalzeichensatz vom ROM ins RAM und verschiebt dann einen 2 KByte großen (zuvor erzeugten) Zeichensatz, der bei 8192 beginnt, in das RAM unter dem ROM ab 53248. Der Groß/Klein-Zeichensatz bleibt erhalten. Man kann also mit SHIFT/C= zwischen zwei Zeichensätzen hin- und herschalten. Will man nur den Originalzeichensatz, so kann man das Verschieben des neuen (der dann natürlich auch nicht da sein muß) mit POKE51234,232 unterbinden. Nun hat man zwei Bildschirme, mit denen man arbeiten kann. Wie man sie einschaltet, ist aus den Zeilen 110 und 115 zu erkennen. Zum normalen Zustand kommt man mit:
POKE56576,151:POKE53272,21:
POKE648,4

## 40147 Bytes free!

Solange man mit den oberen Bildschirmen (ab 49152bzw.50176) arbeitet, kann man den für Basic nutzbaren Speicher vergrößern. Arbeitet man mit Diskette, kann man den Basic-nutzbaren Speicher vergrößern. Arbeitet man mit Diskette, kann man den Basicanfang auf 820 verlegen: POKE820,0: POKE43,53:POKE44,3
Jetzt hat man 1228 Byte mehr Speicher. Mit Datasette kann der Baiscanfang auf 1024 verlegt werden: POKE1024,0:Poke44,4

```
1Ø rem 6Ø-zeichen-programm =======
15 rem (p)    commodore welt      =
2Ø rem ===========================
25 rem (c)    r. schmid-fabian    =
3Ø rem c-64                       =
35 rem ===========================

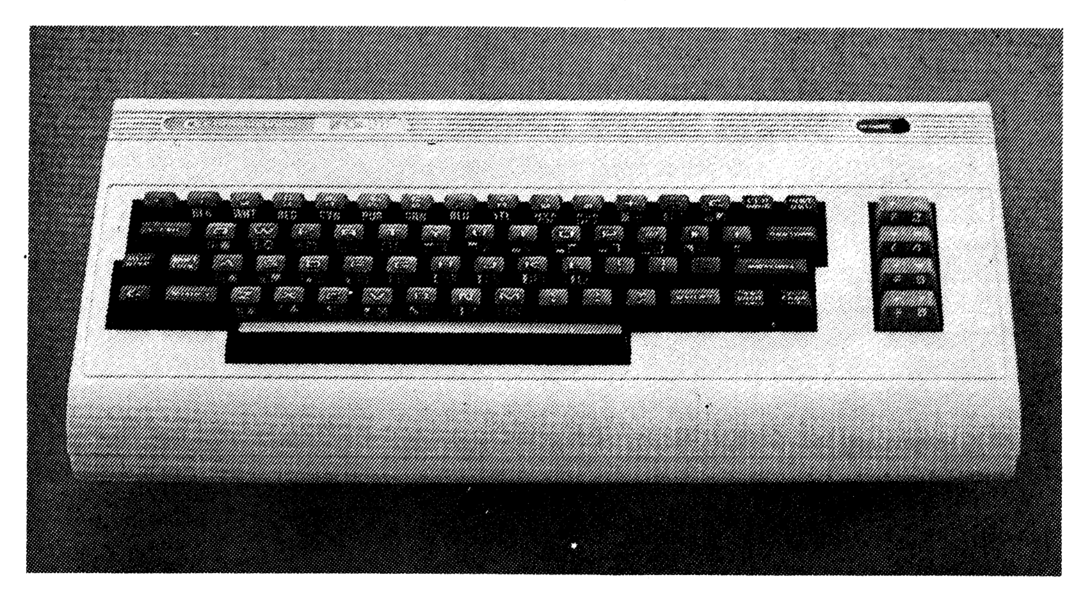
4Ø poke46,5Ø:poke48,5Ø:clr:cl$=chr$
(147)
45 poke5328Ø,Ø:poke53281,Ø:poke646,
3:rem hintergrund schwarz zeichenfa
rbe cyn
5Ø printcl$tab(1Ø)"Neuer Zeichensat
z:"
55 fori=Øto255:poke1424+i,i:next
6Ø rem ** maschinenprogramm ******
65 s=512ØØ:for i=Øto299:readx:pokes
+i,x:next
7Ø rem *** zeichensatz erstellen **
75 r=56334:s=8192:b=1Ø24
8Ø poke53272,24:poker,Ø:poke1,51
85 s=8192:b=1Ø24:for
i=Øto727:readx:pokes+i,x:pokes+i+b,
x/8:next
9Ø s=8192:fori=128Øto1288:pokes+i,1
29:next:pokes+128Ø,255:pokes+1287,2
55
95 poke1,7:poker,1
1ØØ fori=1to5ØØØ:next
1Ø5 rem ** bildschirme loeschen ***
11Ø poke56576,196:poke53272,5:poke6
48,192:print cl$
115 poke56576,196:poke53272,21:poke
648,196:print cl$
12Ø sys512ØØ:rem neuer zeichensatz
nach dØØØ (53248)
125 sys51236:rem bildschirmswitch
13Ø rem text in bildschirme poken *
135 b$="..........................
..................................
...."
14Ø b1=49152:b2=5Ø176:rem startadre
sse der bildschirme
145 rem ascii in bildschirmcode ***
15Ø def fnf(a)=a-161-33*(a<255)-64*
(a<192)-32*(a<16Ø)+32*(a<96)-64*(a<
64)
155 read a$:if a$="stop" then poke1
98,Ø:wait198,1:end
16Ø if a$=" "
then b1=b1+4Ø:b2=b2+4Ø:goto 155
165 a$=left$((a$+b$),6Ø)
17Ø for i=1to6Østep3
175 a=asc(mid$(a$,i,1)):pokeb1,fnf(
a):b1=b1+1
18Ø a=asc(mid$(a$,i+1,1)):pokeb2,fn
f(a):b2=b2+2
185 a=asc(mid$(a$,i+2,1)):if a<>32
then pokeb1,fnf(a)+128:b1=b1+1
19Ø if a=32 then pokeb1,fnf(a):b1=b
1+1
195 next
2ØØ goto155
2Ø5 rem ** maschinenprogramm ***
21Ø data12Ø,16Ø,Ø,132,3,169,2Ø8,133
,4,162,16,169,51,133,1,177,3,72,169
,48,133
215 data1,1Ø4,145,3,2ØØ,2Ø8,239,23Ø
,4,2Ø2,2Ø8,234,76,2Ø5,2ØØ,12Ø,169,1
2Ø,141
22Ø data2Ø,3,169,2ØØ,141,21,3,162,1
99,16Ø,76,173,14,22Ø,41,254,9,144,1
41,14
225 data22Ø,142,4,22Ø,14Ø,5,22Ø,173
,14,22Ø,9,1,168,173,18,2Ø8,234,234,
16,249
23Ø data173,17,2Ø8,16,244,14Ø,14,22
Ø,169,192,141,136,2,173,Ø,221,41,25
2,141
235 dataØ,221,173,24,2Ø8,41,15,141,
24,2Ø8,173,1,221,9,3,141,1,221,88,9
6,Ø,173
24Ø data136,2,73,4,141,136,2,173,22
,2Ø8,73,6,141,22,2Ø8,173,24,2Ø8,73,
16,141
245 data24,2Ø8,173,141,2,2Ø1,4,2Ø8,
6,165,2Ø3,2Ø1,63,24Ø,3,76,49,234,16
9,49
25Ø data141,2Ø,3,169,234,141,21,3,1
73,22,2Ø8,41,248,141,22,2Ø8,173,24,
2Ø8,41
255 data15,9,16,141,24,2Ø8,169,4,14
1,136,2,173,Ø,221,9,3,141,Ø,221,76,
49,234
26Ø dataØ,169,Ø,16Ø,32,133,95,132,9
6,169,Ø,16Ø,4Ø,133,9Ø,132,91,169,Ø,
16Ø,216
265 data133,88,132,89,32,24Ø,2ØØ,16
9,55,133,1,88,96,Ø,Ø,56,165,9Ø,229,
95,133
27Ø data34,168,165,91,229,96,17Ø,23
2,152,24Ø,35,165,9Ø,56,229,34,133,9
Ø,176
275 data3,198,91,56,165,88,229,34,1
33,88,176,8,198,89,144,4,177,9Ø,145
,88,136
28Ø data2Ø8,249,177,9Ø,145,88,198,9
1,198,89,2Ø2,2Ø8,242,96
285 rem**** zeichensatz **********
29Ø data96,144,144,176,176,128,96,Ø
295 dataØ,Ø,24Ø,16,24Ø,144,24Ø,Ø
3ØØ dataØ,128,128,24Ø,144,144,24Ø,Ø
3Ø5 dataØ,Ø,96,128,128,128,96,Ø
31Ø dataØ,16,16,24Ø,144,144,24Ø,Ø
315 dataØ,Ø,24Ø,144,24Ø,128,24Ø,Ø
32Ø dataØ,112,64,224,64,64,64,Ø
```

```
325 dataØ,Ø,96,144,144,112,16,224
33Ø dataØ,128,128,24Ø,144,144,144,Ø
335 dataØ,32,Ø,96,32,32,112,Ø
34Ø dataØ,16,Ø,16,16,16,16,224
345 dataØ,64,64,8Ø,96,8Ø,72,Ø
35Ø dataØ,96,32,32,32,32,112,Ø
355 dataØ,Ø,128,24Ø,176,2Ø8,176,Ø
36Ø dataØ,Ø,16Ø,8Ø,8Ø,8Ø,8Ø,Ø
365 dataØ,Ø,96,144,144,144,96,Ø
37Ø dataØ,Ø,24Ø,144,144,24Ø,128,128
375 dataØ,Ø,24Ø,144,144,24Ø,16,16
38Ø dataØ,Ø,176,64,64,64,64,Ø
385 dataØ,Ø,112,128,96,16,224,Ø
39Ø dataØ,64,224,64,64,64,96,Ø
395 dataØ,Ø,144,144,144,144,24Ø,Ø
4ØØ dataØ,Ø,144,144,144,96,96,Ø
4Ø5 dataØ,Ø,144,144,24Ø,2Ø8,176,Ø
41Ø dataØ,Ø,144,96,96,144,144,Ø
415 dataØ,Ø,144,144,144,112,16,224
42Ø dataØ,Ø,24Ø,48,96,192,24Ø,Ø
425 data112,64,64,64,64,64,112,Ø
43Ø data32,8Ø,64,24Ø,96,2Ø8,24Ø,Ø
435 data112,16,16,16,16,16,112,Ø
44Ø dataØ,32,8Ø,248,32,32,32,32
445 dataØ,Ø,64,248,248,64,Ø,Ø
45Ø dataØ,Ø,Ø,Ø,Ø,Ø,Ø,Ø
455 data64,64,64,64,Ø,Ø,64,Ø
46Ø data8Ø,8Ø,8Ø,Ø,Ø,Ø,Ø,Ø
465 data8Ø,8Ø,248,8Ø,248,8Ø,8Ø,Ø
47Ø data32,112,192,112,24,112,32,Ø
475 data144,144,32,32,64,144,144,Ø
48Ø data96,144,144,96,176,152,1Ø4,Ø
485 data32,64,128,Ø,Ø,Ø,Ø,Ø
49Ø data32,64,128,128,128,64,32,Ø
495 data64,32,16,16,16,32,64,Ø
5ØØ dataØ,144,96,24Ø,96,144,Ø,Ø
5Ø5 dataØ,32,32,248,32,32,Ø,Ø
51Ø dataØ,Ø,Ø,Ø,Ø,32,32,64
515 dataØ,Ø,Ø,112,Ø,Ø,Ø,Ø
52Ø dataØ,Ø,Ø,Ø,Ø,96,96,Ø
525 dataØ,16,32,32,64,64,128,Ø
53Ø data96,144,176,2Ø8,2Ø8,144,96,Ø
535 data32,96,16Ø,32,32,32,112,Ø
54Ø data24Ø,144,16,32,64,128,24Ø,Ø
545 data24Ø,144,16,112,16,144,24Ø,Ø
55Ø data16,48,8Ø,144,248,16,16,Ø
555 data24Ø,128,24Ø,16,16,144,24Ø,Ø
56Ø data24Ø,144,128,24Ø,144,144,24Ø
,Ø
565 data24Ø,144,32,64,64,64,64,Ø
57Ø data96,144,144,96,144,144,96,Ø
575 data96,144,144,24Ø,16,144,96,Ø
58Ø dataØ,Ø,96,Ø,Ø,96,Ø,Ø
585 dataØ,Ø,32,Ø,Ø,32,32,64
59Ø data16,32,64,128,64,32,16,Ø
595 dataØ,Ø,24Ø,Ø,24Ø,Ø,Ø,Ø
6ØØ data128,64,32,16,32,64,128,Ø
6Ø5 data96,144,48,64,64,Ø,64,Ø
61Ø dataØ,Ø,Ø,24Ø,24Ø,Ø,Ø,Ø
615
data96,144,144,24Ø,144,144,144,Ø
62Ø data224,144,144,24Ø,144,144,224
,Ø
625 data112,144,128,128,128,144,112
,Ø
63Ø data224,144,144,144,144,144,224
,Ø
635 data24Ø,128,128,24Ø,128,128,24Ø
,Ø
64Ø data24Ø,128,128,24Ø,128,128,128
,Ø
645 data24Ø,144,128,176,144,144,24Ø
,Ø
65Ø data144,144,144,24Ø,144,144,144
,Ø
655 data112,32,32,32,32,32,112,Ø
66Ø data24Ø,32,32,32,32,16Ø,64,Ø
665 data145,16Ø,192,192,16Ø,176,144
,Ø
67Ø data128,128,128,128,128,128,24Ø
,Ø
675 data144,24Ø,24Ø,144,144,144,144
,Ø
68Ø data144,2Ø8,2Ø8,176,176,144,144
,Ø
685 data96,144,144,144,144,144,96,Ø
69Ø data24Ø,144,145,24Ø,128,128,128
,Ø
695
data24Ø,144,144,144,144,24Ø,24,Ø
7ØØ data24Ø,144,144,24Ø,16Ø,144,144
,Ø
7Ø5
data24Ø,144,128,24Ø,16,144,24Ø,Ø
71Ø data248,32,32,32,32,32,32,Ø
715 data144,144,144,144,144,144,24Ø
,Ø
72Ø
data144,144,144,144,144,24Ø,96,Ø
725 data144,144,144,144,24Ø,24Ø,144
,Ø
73Ø data144,144,96,96,144,144,144,Ø
735 data144,144,144,24Ø,96,96,96,Ø
74Ø data24Ø,16,32,64,128,128,24Ø,Ø
745 rem ***** text *************
75Ø data"123456789Ø123456789Ø123456
789Ø123456789Ø123456789Ø123456789Ø"
755 data" "
76Ø data"......Dies ist ein 6Ø-Zeic
hen-Testprogramm."
765 data " "
77Ø data"......Bitte stellen Sie de
n Kontrast und die Helligkeit"
775 data"......noch etwas ein. Dann
 flimmert es weniger."
78Ø data" "
```

Sechzig Zeichen — 3 —

```
785 data"......Ich
hoffe, es gefaellt Ihnen. "
79Ø data" "
795 data"......Wenn Sie jetzt eine
Taste druecken, koennen Sie"
8ØØ data"......mit dem veraenderten
 Zeichensatz schreiben."
8Ø5 data"......Mit SHIFT/C=
erhalten Sie den normalen"
81Ø data"......Zeichensatz. Mit CTR
L/STOP kommen Sie wieder"
815 data"......zum normalen Bildsch
irm und Zeichensatz."
82Ø data" "
825 data
"........................Rudolf Sch
mid-Fabian"
83Ø data"stop"
```

# IM TEST FÜR DEN VC 20 DAS SUPERTOOL MIT NEUEN BASIC-BEFEHLEN!



**Mit dem VC-SUPERTOOL hat man eines der leistungsfähigsten 'Werkzeuge' zur Programmentwicklung mit dem Commodore VC-20, der in der Grundversion nur mit dem BASIC 2.0 ausgestattet ist, zur Verfügung. Die zusätzlichen Basic-Befehle stehen nach dem Einstecken des Moduls und Einschalten des Commodore VC-20 sofort bereit, oder auch nach Laden des Programms von Cassette oder Diskette in den A000-Bereich eines RAM-Erweiterungsmoduls. Zu beachten ist aber, daß alle VC-20 Module, die den selben Speicherplatzbereich belegen, sich gegenseitig ausschließen, oder die Module können beschädigt werden. Wenn Sie eine Steckplatzerweiterung benutzen, müssen z.B. Spielmodule, die in diesem Bereich arbeiten, herausgezogen oder abgeschaltet werden.**

Die Zusatzbefehle werden in vier Gruppen unterteilt:

(A) Programmierhilfen
(B) Schnelle Kassettenoperationen
(C) Druckerausgabe über USER-Port
(D) Strichcode-Leser

Alle Befehle können auch in der vom Computer erlaubten Kurzform eingetippt werden. Es wird dazu der erste Buchstabe normal und der zweite geshiftet eingegeben. Bei der Eingabe von 'LIST' erscheinen die Befehle dann wieder in voller Länge. Bei manchem Befehl müssen die Parameter nicht unbedingt eingegeben werden. VC-Supertool setzt dafür sinnvolle Werte selbständig ein.

## (A) PROGRAMMIERHILFEN:

AUTO sorgt für automatische Zeilennumerierung, mit variabler Startnummer und Schrittweite.
AUTO120,5 beginnt mit Zeilen-Nr. 120 und fährt in 5er Schritten fort
AUTO 1000 beginnt mit Zeilen-Nr. 1000 und fährt in 10er Schritten fort
AUTO beginnt mit Zeilen-Nr. 100 und fährt in 10er Schritten fort.
Wenn Sie während des Programmierens Zeilennummern überspringen wollen, überschreiben Sie die automatisch erzeugte Zeilennummer mit der gewünschten Zeilennum-

mer. Die vorher gewählte Schrittweite wird dann von dieser neuen Zeilennummer ab weitergeführt. AUTO wird wieder ausgeschaltet, wenn nach der zuletzt erzeugten Zeilennummer RETURN gedrückt wird.
RENUM numeriert alle Zeilen eines im Speicher stehenden BASIC-Programms neu.
RENUM5,2 beginnt mit Zeilennummer 5 und fährt in 2er Schritten fort
RENUM50 beginnt mit Zeilennummer 50 und fährt in 10er Schritten fort
RENUM beginnt mit Zeilennummer 100 und fährt in 10er Schritten fort.
RENUM berechnet alle Bezüge (GOTO/GOSUB/LIST/RUN) innerhalb des Programms auf andere Zeilennummern nach. Wenn eine Zeilennummer, auf die gesprungen werden soll, nicht vorhanden ist (fehlt im Programm), so setzt RENUM dafür die Zeilennummer 63999 ein und macht so auf den Fehler aufmerksam. Würden beim Umnumerieren Zeilennummern über 63999 entstehen wird '? OVERFLOW ERROR' gemeldet und RENUM wird nicht ausgeführt.

DEL (DELETE) löscht mehrere Zeilen gleichzeitig im Programm.
DEL 160–225 löscht alle Zeilen von 160 bis einschließlich 225
DEL180– löscht das gesamte Programm ab Zeile 180
DEL –126 löscht alle Zeilen bis 126 einschließlich.

TRACE arbeitet wie RUN und zeigt im rechten oberen Bildschirmfeld die Zeilen an, die das Programm durchläuft. So können leicht fehlerhafte Programmierungen gefunden werden. Auch hat es sich bei der Analyse unbekannter Basic-Programme sehr bewährt. Will man erst bei einer bestimmten Zeilennummer starten, so wird diese Zeilennummer hinter den TRACE-Befehl gesetzt.
TRACE läuft nur so lange, wie irgendeine Taste gedrückt wird. Durch kurzes Tippen kann auch Einzelschritt-Betrieb durchgeführt werden.

HELP zeigt nach einer Fehlermeldung, in welcher Zeile an welcher Stelle der Zeile der Fehler zu suchen ist.

DUMP gibt nach einem Programmlauf an, wie die Variablen (keine Felder) belegt werden.

FIND sucht Zeilen, in denen ein vorgegebener Suchbegriff vorkommt.
FIND IF listet alle Zeilen, in denen ein IF-Befehl steht.
Dabei kann der Suchbegriff auch Teil eines Wortes sein. Beispiel: gesetzt.
FIND "set" listet alle Zeilen mit dem String 'set'.
FIND kann übrigens auch nur in einem Teil des Programms suchen. Man gibt die Z.-Nummern wie bei LIST ein. Zu beachtender Unterschied: Ist der Suchbegriff ein Basic-Befehl, so muß dieser Befehl und die Zeilennummer durch ein Komma getrennt werden. Ist der Suchbegriff ein String, übernimmt das ' " ' diese Trennung.

## (B) KASSETTENBEFEHL

PUT speichert ein Programm über den Kassettenrecorder mit ca. zehnfacher Geschwindigkeit gegenüber dem normalen SAVE ab. Dabei wird zuerst 10 Sekunden gewartet, bis das Vorspannband der Kassette vorbeigelaufen ist.
PUT"xyz" speichert ein Programm mit dem Namen 'xyz'
PUT* wirkt wie PUT, wartet aber nicht 10 Sekunden. So können alle Programme nach dem ersten auf der Kassette ohne die Wartezeit abgespeichert werden.

GET liest Programme, die mit PUT gespeichert wurden, ein.
GET"xyz" sucht und liest dann das Programm mit dem Namen 'xyz' ein.

COMP (compare) ist die schnelle Version von VERIFY.

MERGE ermöglicht das Ankoppeln von Programmen, die mit PUT auf einer Kassette geschrieben wurden, an ein Programm im Hauptspeicher. So können z.B. schon vorhandene Subroutinen in neue Programme übernommen werden.
Zuerst wird mit GET das erste Programm eingeladen und dann mit MERGE das zweite Programm angehängt. Es empfiehlt sich, mit RENUM vorher dafür zu sorgen, daß die Zeilennummern beider Programme sich nicht überschneiden.

## (C) DRUCKERAUSGABE

VC-Supertool gestattet den Anschluß eines 8-Bit-Druckers mit sogenannten Centronics-Schnittstelle an den Userport des VC-20, ohne daß ein teures, spezielles Interface dafür angeschafft werden muß. Es ist nur ein Kabel mit Userport- und Centronics-Stecker erforderlich.
Die Datenleitungen des Userport B werden mit den entsprechenden Dateneingängen des Druckers verbunden. CB2 ist mit Data-Strobe, CB1 mit Acknowledge zu verbinden. Es ist zweckmäßig, ein 14- bis 16adriges Kabel zu nehmen und vier bzw. sechs Adern als Masseleitungen zu verwenden.
Zum Beispiel der ITOH Centronics-Drucker 8510A. Für ihn ist ein Zeichengenerator-Rom mit sämtlichen Zeichen (auch Grafik) des VC-20 erhältlich. Da auch Einzelnadel-Ansteuerung möglich ist, eignet er sich zum hochauflösenden Plotten, Strichcode-Drukken etc.
Der Drucker kann über vier verschiedene Primäradressen (16 bis 19) angesprochen werden. Damit wird festgelegt, in welchem Modus die Zeichen ausgegeben werden.

OPEN 1,16 ASCII-Modus (z.B. zum Plotten)
OPEN 1,17 LIST-Modus: Steuerzeichen werden druckbare Zeichen
OPEN 1,18 TEXT-Modus: Groß-/Kleinschreibung
OPEN 1,19 GRAFIK-Modus (nur mit neuem Zeichengenerator)
Beispiel: OPEN1,17: CMD1:LIST druckt ein Listing eines Programmes. Danach muß durch PRINT#1:CLOSE1 der CMD-Modus ausgeschaltet werden.
Über die allgemeine Verwendung der Befehle OPEN, CMD, PRINT# und CLOSE siehe auch VC-20 Handbuch und entsprechende Programmierhandbücher.

## (D) STRICHCODELESER

BAR (Barcode=Strichcode) ist ein Hilfsprogramm, mit dem man Strichcode-Programme z.B. aus Zeitschriften direkt in den Basicspeicher einlesen kann. Der gelesene Text erscheint gleichzeitig auf dem Bildschirm. Der Lesestift wird an den Kassetten-Port angeschlossen.

*Erhältlich bei:*
*Mukra Datentechnik*
*Schöneberger Str. 5*
*1000 Berlin West 42*
*Tel.: 030/752 91 50*
*Preis: DM 99,–*

Diese Befehlserweiterung ist für editiermüde Anwender eine durchaus brauchbare Programmierhilfe. Es werden Funktionen angeboten, die für jeden Computer zur Grundausstattung des BASIC gehören sollten, wie es zum Teil im BASIC 3.5 und 7.0 erfüllt werden. Vielleicht können sich die Computerhersteller einmal entschließen, komfortable, gebrauchsfertige und gebrauchstüchtige BASIC-Versionen anzubieten.

# MÜLL-ABFUHR FÜR DEN C 64

**Räumen Sie doch einfach Ihren Speicher mal kräftig auf. Wie das geht, steht auf dieser Seite.**

Die Mängel der INPUT-Routine des C64 sind bekannt: Kein Komma oder Doppelpunkt in der Eingabe und nur 80 Zeichen, obwohl der Eingabepuffer wie beim VC20 88 Zeichen faßt. Will man bei der Eingabe alle Zeichen erlauben, geht man normalerweise wie folgt vor:

10 GET A$: IF A$= " " THEN 10
20 B$= B$+ A$: PRINT A$;: GOTO 10

Wobei dann B$ im Programm weiterverarbeitet wird. Der entscheidende Nachteil der Methode liegt in der ‚hausinternen' Müllabfuhr ‚sprich' garbage collection' des Computers. Jedesmal, wenn B$ verändert wird, so wird er neu im Speicher abgelegt und nur seine Anfangsadresse geändert. Ist der Speicher voll, muß erst einmal der ganze Müllstring beseitigt werden und das kann unter Umständen mehrere Minuten dauern, abhängig davon, wieviel andere Variablen definiert und verändert wurden.

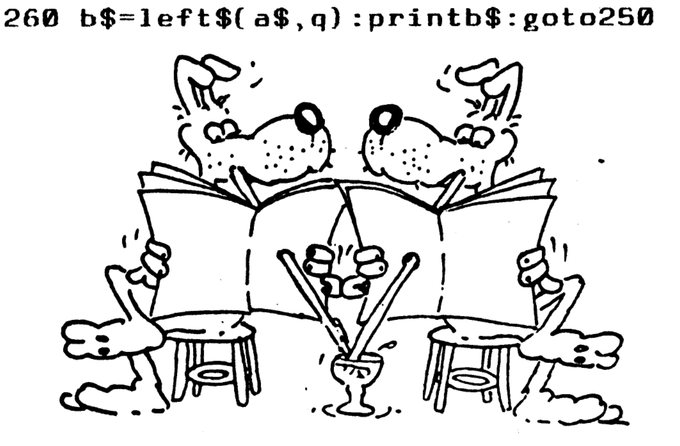
Hier nun eine völlig andere Methode: In Zeile 190 wird aus dem Inhalt der Speicherstellen 45/46 der Anfang der Variablen errechnet. Da vor der Definition von A$ ein CLR durchgeführt wird, steht A$ als erste Variable im Speicher an der Stelle VA. Bei VA steht daher 65 (CHR$-Code von A). In VA+1 steht 128, wodurch A als Stringvariable gekennzeichnet wird. In VA+2 steht 84 (Länge von A$) und in VA+3/VA+4 steht das LO/HI-Byte der Adresse, bei der der String anfängt. Dies gibt uns die Möglichkeit, A$ durch überPOKEn der alten Werte zu verändern.

Nun gibt es noch zwei sehr nützliche Betriebssystemroutinen. Wird beim Schreiben auf dem Bildschirm die 40te Spalte erreicht, so wird eine Folgezeile eröffnet, das heißt, daß festgelegt wird, daß diese beiden Bildschirmzeilen zu einer ‚logischen' Zeile zusammengehören.

Das gleiche erreicht man mit SYS59749. SYS 42336 ruft die Eingaberoutine auf, welche auf RETURN wartet und dann die logische Zeile in den Eingabepuffer schreibt, auf der der Cursor gerade steht. Jetzt braucht man nur noch die Werte aus dem Eingabepuffer 512-600, (am Ende steht immer eine Null) in den Speicherbereich von A$ zu übertragen und dann in B$ den interessierenden Teil aufnehmen, um ihn dann im Programm weiterzuverarbeiten. Nun hat man eine Eingaberoutine zur Verfügung, die bis zu 84 Zeichen ohne Einschränkung akzeptiert. Werden mehr als 84 eingetippt, so werden nur 84 verwendet.

Bei mehr als 88 steigt das Programm mit ‚STRING TOO LONG ERROR' aus.

```
Inputroutine
10 rem inputroutine 84 zeichen ===
20 rem mit wenig muell ===========
30 rem (p) commodore welt        =
40 rem ==========================
50 rem (c) r. schmid-fabian      =
60 rem c-64                      =
70 rem ==========================
100 rem *************************
110 rem *  a$ definieren     ****
120 rem *************************
130 printchr$( 147)chr$( 14);
140 a=59749:rem platz fuer folgezei
le
150 b=42336:rem eingaberoutine
160 sysa:sysa:print"a$="chr$( 34);:f
or i=1to84:printchr$( 160);:next
170 print:print:print:print"goto190
"
180 clr:poke631,19:poke632,13:poke6
33,13:poke198,3:end:rem
tastaturpuffer
190 va= peek( 45)+256*peek( 46):rem v
ariablen anfang
200 sa= peek( va+3)+256*peek( va+4):r
em start a$
210 rem *************************
220 rem * string einlesen    ****
230 rem *************************
240 a=59749:b=42336:printchr$( 147)
250 sysa:sysa:sysb:for q=0to84:w=pe
ek( 512+q):if w then pokesa+q,w:next
260 b$=left$( a$,q):printb$:goto250
```



## Nur Mut!

Außer zur Zeichenketteneingabe in Programmen kann man die Routine auch verwenden, um sich ein Autonumber-Programm (automatische Vorgabe der Zeilennummern) zu schreiben, das 84 Zeichen akzeptiert. Wenn Sie die Karteikarten aus der CBM-Revue 2/86 dazunehmen, haben Sie alles, was man dafür braucht. Auch 88 Zeichen sind möglich. Hierzu muß die 84 in VA+1 auf 88 erhöht und die SA um 4 erniedrigt werden. Nur Mut. Nur selberprogrammieren macht schlau.

*(rSf)*

## WO GIBT ES DAS ROBCOM TURBO...

In Ihrer Zeitschrift COMMODORE-WELT, Ausgabe März '86 berichten Sie über ein Modul Robcom Turbo 50. Habe hier bei den Händlern nachgefragt, aber keiner konnte es mir besorgen. Daher die Bitte an Sie, mir einen Quellennachweis zu übersenden, wo ich das Modul erwerben kann.
**Klaus Grettern**
**Stolberg**

## ...UND DIE KASSETTEN-EINSTELL-HILFE?

In Heft 3/86 wurde das Robcom Turbotool Modul besprochen. Leider fehlt jede Angabe über eine Bezugsquelle.
In Heft 4/86 wurde eine Einstellhilfe für den Kassettenrecorder vorgestellt, aber leider ebenfalls ohne Bezugsnachweis.
**Rolf Schmitz**
**Büren**

*Stellvertretend für die vielen Anfragen, die uns dazu erreichten, hier die Adressen der Hersteller, bzw. Importeure, bei denen auch ein Händlerverzeichnis erfragt werden kann:*
*Robcom Turbo 50:*
*Mastertronic GmbH*
*Kaiser Otto Weg 18*
*D-4770 Soest*

*Aackosoft Head Alignment Kit*
*Aackosoft International*
*P.O. Box 3111*
*2301 DC Leiden/ Niederlande*

## WIE IST DAS MIT DEM E.V.?

Mit großer Freude habe ich in CW 2/86 den Artikel „So gründen Sie einen Computerclub" gelesen, da ich schon seit einiger Zeit einen solchen ins Leben rufen möchte.
Nun laufen die Vorbereitungen auf Hochtouren. Mitglieder sind schon gefunden und wir haben auch so ziemlich die gleichen Interessen.
Das erste Problem stellt sich jedoch, wenn es um das Eintragen des Clubs geht. Aus Neugierde gingen zwei von uns zum städtischen Rathaus, um dort Informationen zu bekommen.
Dort erzählte man ihnen, daß sie mit diversen Unterlagen einen Antrag beim Amtsgericht stellen müßten.
Da über die „diversen Unterlagen" keine weiteren Auskünfte gegeben werden konnten, wende ich mich nun an Sie, um Näheres über das Eintragen eines Clubs zu erfahren.
Nun komme ich zu Problem Nr. 2:
In obengenanntem Artikel kam unter anderem zu Frage 2 die Antwort, man könne Schulungen durchführen.
Nun haben wir uns überlegt, daß man, um Fahrten zu Messen usw. zu finanzieren, Kurse in Basic und Pascal abhalten könnte.
Räume sind vorhanden, Computeranlagen auch, ebenso vollständige Kursunterlagen usw.
Um diese Kurse zu testen, haben wir sie je zweimal mit sehr zufriedenstellendem Erfolg an völlig computerunerfahrenen Personen, wie z.B. Eltern, Verwandten und Freunden, durchgeführt.
Alle sind nun in der Lage, Probleme in Basic und Pascal zu lösen und teilweise sogar komplexe Programme zu schreiben. Allerdings hat die Sache einen Haken: Jemand sagte uns, daß man doch eigentlich eine Lehrgenehmigung haben müsse, um solche Kurse gegen Bezahlung durchzuführen. Nun möchten wir wissen, inwiefern solch ein Kurs legal wäre und wann man Steuern etc. bezahlen muß.
**Erik Jacobi**
**Radevormwald**

*1. Wenn es unbedingt ein eingetragener Verein werden soll, so haben Sie eine Menge Papierkram vor sich: Es muß eine Vereinssatzung erstellt werden, Vorstand, Kassenwart, Schriftführer usw. gewählt werden, deren turnusmäßige Wiederwahl bestimmt werden und so weiter...*
*Um hierfür genaue Informationen zu erhalten, wenden Sie sich am besten schriftlich (!) an Ihr Amtsgericht und fordern die erforderlichen Unterlagen an, dort kann man Ihnen mit Sicherheit mehr sagen als wir.*
*2. Wollen Sie Ihre Schulungen öffentlich und gegen Bezahlung durchführen, benötigen Sie einen Gewerbeschein, den Sie von Ihrem zuständigen Rathaus erhalten. Für diese Schulungen müssen dann natürlich auch die betreffenden Steuern abgeführt werden.*
*Setzen Sie soch doch auch einmal mit anderen Computerclubs in Verbindung und erfragen Sie dort, welche Erfahrungen gemacht wurden und welche Auflagen zu erfüllen sind.*

**LESERBRIEFE UND FRAGEN SIND UNS STETS WILLKOMMEN. WIR BEANTWORTEN SIE ENTWEDER DIREKT ODER AUF DER LESERBRIEFSEITE**

## SCHWIERIGKEITEN MIT SEQUENTIELLEN DATEIEN

Mein Problem, bei dem Sie mir sicher helfen können:
Ich besitze den PC 128 und die Floppy 1571. Schwierigkeiten habe ich beim Erstellen sequentieller Dateien. Sobald ich eine Datei eröffne mit --DOPEN# usw. -- arbeitet die Floppy kurz, dann hört man nichts mehr und die grüne Kontrollampe brennt (Dauerlicht). Hinweisen möchte ich noch darauf, daß ich ganz unterschiedliche Programme (sequentiell) versucht habe, immer mit dem gleichen Ergebnis, so daß man wohl einen Programmfehler ausschließen kann. Die Fehlerabfrage --Print DS$-- bestätigt mir aber immer, daß bei der Floppy alles in Ordnung ist. Da es in Bielefeld keine speziellen Commodore-Läden gibt, wende ich mich an Sie mit der Bitte, mir Ihre Meinung dazu mitzuteilen.
**Wilhelm Kornfeld**
**Bielefeld**

*Versuchen Sie es ganz einfach mit den Befehlen des 64ers, denn diese funktionieren auch im 128er Modus. Zum Erstellen sequentieller Files auf Diskette also:*
*OPEN 1,8,1,"............,S, W":CMD 1:LIST*
*Die Punkte stehen hier für den Namen, mit dem das zu erstellende File auf Diskette geschrieben wird (bis zu 16 Zeichen). Wenn sich der Cursor wieder auf dem Bildschirm meldet, muß CLOSE 1 eingegeben werden, um den Diskettenkanal wieder zu schließen. Wundern Sie sich nicht, wenn beim nächsten Diskettenzugriff die Fehlermeldung >DEVICE NOT PRESENT< erscheint. Nach dem Erstellen von sequentiellen Dateien*

*braucht der 128er immer eine zweite Aufforderung, um die Diskettenbefehle auszuführen.*

## ZAHLENBLOCK NICHT BENUTZBAR!

Vor kurzem habe ich mir den C128 gekauft. Nachdem ich die Bedienungsanleitung gelesen hatte, war ich in einem Punkt sehr enttäuscht: Im 64er-Modus sind viele Tasten, wie auch der Zahlenblock, nicht benutzbar. Nun meine Frage an Sie: Gibt es eine zuverlässige Softwarelösung, um im 64er-Modus auch die Tasten des 128er-Modus benutzen zu können? Denn dies wäre gerade bei DATA-Eingaben sehr hilfreich.

**Michael Dumont**
**4242 Rees-3**

*Leider, leider. Auch uns ist in dieser Richtung nichts bekannt, die einzige spezielle 128er Taste, die auch im 64er Modus benutzt werden kann, ist die zum Umschalten zwischen ASCII und DIN Tastatur. Eine Einschränkung hierbei: Natürlich kann der C 64 nicht alleine mit dieser Taste die deutschen Umlaute darstellen, welche auf der Tastatur des 128 PC angegeben sind. Durch die Umschaltung erhalten Sie lediglich ein schmaleres Schriftbild und einige Sonderzeichen wie etwa das Paragraphenzeichen. Ob dieses Problem lösbar ist, softwaregesteuert lösbar ist, scheint zudem etwas fraglich, da ja zwischen zwei völlig verschiedenen Systemen umgeschaltet werden muß. Ein Beispiel: Zwar ist es ja bekanntlich möglich, vom 128er Modus in den des 64er mit dem Befehl >GO 64< umzuschalten, umgekehrt funktioniert das jedoch nur durch Ausschalten bzw. RESETEN. Eine Möglichkeit für Sie: Bei Exbasic II haben Sie einige Zusatzfunktionen wie z.B. HELP, außerdem sind im Handel externe Zehnertastaturen für den C 64 als Bausatz oder Komplettgerät erhältlich, die auch am 128er im C 64 Modus verwendet werden können.*

## DENKT MEHR AN DIE ANFÄNGER!!!

Zu Weihnachten bekam ich einen Commodore 16 geschenkt. Ich gehöre der Generation an, die nun schon auf die 40 zugeht, und (vielleicht auch technisch nicht allzu begabt) sich schwer tut mit dem „computern". Mir fehlt es als Hausfrau auch an jeglichen Vorkenntnissen.
Doch mir geht es nicht alleine so. Viele Elternteile – halt auch „einfache Leute" – würden gerne mit ihren Kindern mitmachen. Ich kenne einige Mütter, die gerne „im Gespräch" mit ihren Kindern bleiben würden. Doch für diese Gruppe der Computer-Fans fehlt leider Fachliteratur. Echte Lektüre für den totalen Anfänger.
Im mitgelieferten Handbuch wird auch ausdrücklich auf genauere Literatur verwiesen, die ich aber gerade für den Commodore 16 nirgendwo (Kaufhäuser, Geschäfte) finden kann.
Manches verstehe ich aus dem Handbuch, vieles nicht. Für viele Funktionen (Grafik-Teil) sind Beispiel-Programme eingegeben – aber keine Erklärungen dazu. Ein Programm für eine Kurve – aber will ich die Kurve verändern, muß ich das Programm verändern, aber ich weiß nicht, welche Zeile – und Ähnliches!
Von den Schwierigkeiten mit den "Neudeutschen" Fachbegriffen ganz zu schweigen.
Auch Ihre Zeitschrift war in dieser Hinsicht für meine Bekannten und mich eine herbe Enttäuschung. Wir haben die Nr. 2/86 gekauft und genau "0" Artikel gefunden, die bei unserem Wissens-Stand einsetzbar gewesen wären. (Außer dem Thema Club-Gründung – aber die meisten verlieren am ganzen Thema Computer schon wieder die Lust!)
Aus diesem Grunde habe ich diesen Brief geschrieben, denn auch Anfänger sind halt potentielle Kunden, wenn man sie bei der Stange halten kann. Vielleicht können Sie mir einen Tip für Anfänger-Lektüre für den Commodore 16 geben – ich wäre Ihnen sehr dankbar.
In jedem Fall würde ich aber vorschlagen, die Anfänger generell nicht zu vergessen. Sonst verstärkt sich der Trend, den ich im Verwandtschaftskreis bemerke = Leute unter 30 Computer-Fans/Leute über 30 Computer-Gegner und Verteufler. Gegner deshalb, weil uns Menschen die Trauben niemals zu hoch sondern immer zu bitter sind.

**Jutto Kock**
**Berlin**

*Liebe Frau Kock, die im Handbuch aufgeführte Literatur werden Sie vermutlich bei dem Händler erhalten, bei welchem auch Ihr Computer gekauft wurde (falls er gut sortiert ist). Wenn Sie dort keinen Erfolg haben, wenden Sie sich direkt an Commodore, dort müssen diese Bücher ebenfalls erhältlich sein. Im freien Handel erhältlich ist außerdem folgendes Buch, welches sich speziell mit dem C 16/116 auseinandersetzt und auch eine grundlegende Einführung in das "Computern" enthält:*
*Basic-Wegweiser für den C 16, 116, Plus/4*
*Ekkehard Kaier*
*ISBN-Nr. 3-528-04337-7*
*Vieweg Verlag, Wiesbaden*
*45,– DM*
*Außerdem führen inzwischen die meisten großen Buchverlage entsprechende Titel, vom Grundwissen bis zu Spezialanwendungen, genauere Informationen erhalten Sie sicherlich von jeder Buchhandlung.*
*Zu Ihren Schwierigkeiten mit der Verständlichkeit von Fachbegriffen: Wir haben eigentlich bis jetzt die Erfahrung gemacht, daß auch von Computerneulingen unsere Zeitschrift gelesen und verstanden wird. Wir bemühen uns, die Artikel selbst für Computerlaien so verständlich wie möglich zu halten, was auch durch positive Reaktionen der Leser bestätigt wird.*

### Electronic Swiss Mailing – Computer Club Switzerland

Der ESM-Computer Club Switzerland (ESM-CCS) ist vor ein paar Monaten aus der DATA 64 und dem ESM-CCR hervorgegangen. Der Club besitzt zur Zeit etwa 150 Mitglieder. Wir sind nicht an ein bestimmtes Computersystem gebunden und haben auf diese Weise Computeruser aus allen Sparten in uns vereint. Der Erfahrungsaustausch steht für uns an erster Stelle der Ziele, die sich der ESM-CCS gesetzt hat.
Um den Kontakt unter Computerfreunden zu verbessern, sind wir Betreiber einer Mailbox. Außerdem erscheint monatlich unsere 20 Seiten umfassende Zeitschrift Computer Times, in der wir Tips & Tricks, News und Soft & Hardware behandeln. Im Jahresbeitrag von 40,– Franken (30,– Schüler) ist die Zeitschrift enthalten. Infos sind anzufordern bei:
ESC-CCS
Postfach
4125 Riehen
Schweiz

# 128 PC HARD-COPY AUF TASTEN-DRUCK

Der PC 128 hat einen eingebauten Maschinensprachemonitor und einen Mini-Assembler sowie leicht programmierbare Funktionstasten. Mit diesen Elementen ist es leicht möglich, eine 80 Zeichen Hardcopyroutine auf eine Funktionstaste zu legen.

Diese Routine ist in Maschinensprache geschrieben und ist nur ca. 500 Bytes lang. Sie steht in der Bank 15 (hex F) und dürfte somit sicher vor etwaigen Überschreibungen sein. Der 80 Zeichenvideocontroller verfügt über einen eigenen 16 kB-Speicher, diesen kann man über die Adressen $d600 und $d 601 ansprechen bzw. auslesen. Über diese Adressen muß nun Byte für Byte ausgelesen werden.

Das wäre für ein Basic-Programm eine zu mühselige und vor allen Dingen langsame Angelegenheit. Um schnell zu sein, wird also in 'Assembler' (Übersetzer) programmiert. Der PC 128 hat einen sogenannten Direktassembler, das heißt, daß die Befehle sofort in Maschinencode übersetzt werden. Somit ist es also nicht möglich, ein sogenanntes Quellcode-listing zu erstellen. Dies ist auch der Grund für das etwas seltsam anmutende Listing. Das abgedruckte Listing stellt den disassemblierten (rückübersetzten) Speicherinhalt dar. Die linke 'Zahlenreihe' sind die Speicherplatznummern, die nachfolgende 3-spaltige Reihe stellt den Speicherinhalt in hexadezimaler Schreibweise dar. Die beiden rechten Spalten enthalten ebenfalls den Speicherinhalt, aber in m-nemonischer Schreibweise und in dieser Form muß auch das Programm eingegeben werden. Denn nur diese Sprache versteht unser

```
M12F0 1430
>012F0 2A 43 4F 4D 4D 4F 44 4F 52 45 20
>01300 85 FB 86 FD A9 00 85 FA 85 FC A9
>01310 20 BA FF 20 C0 FF A2 01 20 C9 FF
>01320 A9 00 A8 AA 8E 0D 14 8C 0E 14 A2
>01330 14 E8 A5 FC 20 01 14 A2 1F 8E 00
>01340 FB AD 01 D6 AE 0D 14 AC 0E 14 8D
>01350 14 10 05 A9 11 20 D2 FF 8E 0D 14
>01360 A5 FB 20 01 14 E8 A5 FA 20 01 14
>01370 2C 00 D6 10 FB AD 01 D6 AE 0D 14
>01380 14 18 AD 0D 14 10 09 49 80 48 A9
>01390 85 FE 29 3F 06 FE 24 FE 10 02 09
>013A0 C9 22 D0 16 8E 0D 14 A2 00 BD 0F

>013B0 E0 08 D0 F5 AE 0D 14 4C BD 13 20
>013C0 F4 A9 92 20 D2 FF A9 91 20 D2 FF
>013D0 85 FA A5 FB 69 00 85 FB 18 A5 FC
>013E0 FD 69 00 85 FD C8 C0 50 F0 03 4C
>013F0 D2 FF A0 00 E8 E0 19 D0 EF 20 CC
>01400 FF 8E 00 D6 2C 00 D6 10 FB 8D 01
>01410 28 43 29 20 34 2F 31 39 38 36 20
>01420 50 45 54 45 52 20 42 41 53 43 48
>01430 2A 43 4F 4D 4D 4F 44 4F 52 45 20

SYS4864
READY.
```

```
57 45 4C 54 2A:
01 A2 04 A0 FF:
A9 0D 20 D2 FF:
12 A5 FD 20 01:
D6 2C 00 D6 10:
0D 14 18 AD 0D:
8C 0E 14 A2 12:
A2 1F 8E 00 D6:
AC 0E 14 8D 0D:
12 20 D2 FF 68:
80 70 02 09 40:
14 20 D2 FF E8:

D2 FF A9 00 85:
18 A5 FA 69 01:
69 01 85 FC A5:
24 13 A9 0D 20:
FF A9 01 4C C3:
D6 60 20 4F FF:
43 57 20 26 20:
00 00 00 00 00:
57 45 4C 54 2A:
```

Prozessor, auch Basic wird im Interpreter in diesen Code übersetzt und das kostet viel Zeit. Um das Programm eingeben zu können, muß der Monitor mit der F8-Taste eingeschaltet werden. Er meldet sich wie im Listing angezeigt. Mit dem 'A-Befehl' (assemblieren) können nun alle Anweisungen an den Prozessor übergeben werden. Die erste Eingabe sieht so aus: A F1300 STA $FB 'RETURN'.

Sofort nach dem Drücken der Return-Taste wird die Anweisung

## Für Hiresgraphik nicht geeignet

übersetzt und der Rechner gibt Ihnen den nächsten Speicherplatz an. (Ähnlich der automatischen Zeilennummerierung in Basic). Sie brauchen also nur die beiden rechten Reihen einzugeben. Also ist die nächste Eingabe STX $FD 'RETURN', danach LDA #$00 usw. Ihre letzte einzugebende Zeile ist JSR $084F, die beiden anderen mit den Fragzeichen brauchen Sie nicht einzugeben. Statt dessen drücken Sie zweimal 'RETURN', es wird keine Speichernummer mehr ausgegeben und Sie können nun das Programm mit:

S'HARDCOPY, 08, F1300 F1410

auf Disk abspeichern. Wenn Sie alles richtig gemacht haben, so können Sie bereits jetzt eine Hardcopy Ihres Bildschirmes drucken lassen. Sie geben nur 'SYS 4864' und 'RETURN' ein. Ihr Drucker, wenn eingeschaltet, muß jetzt losrattern, allerdings nur dann, wenn Sie vorher den Monitor mit 'X' verlassen haben. Um das Programm von Disk in den Speicher zu laden, brauchen Sie nicht mehr den Monitor, sondern Sie können mit BLOAD'-HARDCOPY' die Routine laden. Wenn Sie das Programm mit einer Funktionstaste aufrufen wollen, so müssen Sie eine der Tasten z.B. F1 folgendermaßen belegen:

KEY F1, 'SYS 4864'+CHR$(13)

Wenn Sie jetzt die F1-Taste drükken, wird sofort eine Hardcopy erstellt. Das geht allerdings nur bei Text und Blockgraphik, für Hiresgraphik ist diese Routine nicht geeignet. Aus einem Programm heraus können Sie die Hardcopy mit SYS DEC ('1300') ebenfalls starten.

*Peter Basch*

# BLOCKS USE ON TRACK 18

Dieses Programm läuft auf dem Commodore 64 in Verbindung mit einer Floppy 1541. Tippen Sie dieses Listing ein und vergewissern Sie sich, daß Sie das Programm abgespeichert haben. Starten Sie nun das Programm. Der Computer wird Sie auffordern, die zu bearbeitende Diskette einzugen. Ist dies geschehen, so überprüft der Computer, ob die Diskette eventuell noch schreibgeschützt ist Danach fragt er Sie, ob Sie ein "Validate" machen wollen. Dies ist sehr zu empfehlen, wenn mit der Diskette nach dem letzten "Validate" noch einmal gearbeitet wurde.
Als nächstes erwartet er die Anzahl der Blocks, die Sie frei gemacht haben wollen. Nach der Eingabe dieser Zahl (min. 0 max. 17) beginnt er zu arbeiten. Stehen nicht genügend Blocks zur Verfügung, so meldet er sich mit einer Fehlermeldung wieder. Wählen Sie nun unter den drei Möglichkeiten:

1) Das Programm beenden
2) Weniger Blocks besetzen

oder

3) Das Programm neu starten.

Sind genügend Blocks vorhanden, so arbeitet er und zeigt dabei an, was er gerade macht.
Der Computer führt am Schluß noch ein "Validate" aus und fragt Sie, ob Sie das Programm noch einmal brauchen. Ist dies nicht der Fall, so wird das Programm beendet und die benötigte Zeit ausgedruckt.
Wartet der Computer auf eine Eingabe, so können Sie dann mit "$" die Directory auf den Bildschirm bringen. Durch Druck auf die "Return"-Taste kehren Sie wieder ins Menü zurück.

## Zeile/Beschreibung

1180–1220
Felder aufbauen
1260–1330
Uhr auf Null stellen
1500–1540
Abfrage auf Schreibschutz
1690–1780
Anzahl der freien Blocks wird ermittelt
1790–1910
Fehlermeldung bei zu wenigen Blocks
1960–2070
Sucht in der Directory nach PRG- oder SEQ-File, um diesen dann umzukopieren
2140–2310
Merkt sich die zu verarbeitenden Blocks
2320–2580
Kopiert die richtigen Blocks um
2590–2820
Stellt die Directory wieder her
3380–3520
Beseitigt auf Wunsch den Schreibschutz
3540–3750
Geht die Blocks der Directory nach, markiert diese in der BAM (falls dies noch nicht geschehen ist)
3770–3960
Maschinenroutine,um die Directory einzulesen

## Was macht das Programm?

Eine Diskette ist nach der Formatierung in 35 Tracks eingeteilt. Auf Track 18 befinden sich 19 Sectoren. Auf Sector 0 steht der Name der Diskette und die BAM. Die Nameneinträge der Files startet auf Sector 1.
Für je acht Einträge braucht man einen eigenen Block. Daher besteht die Möglichkeit, 144 Files (18 Sectoren a 8 Einträge) pro Diskette zu haben. Da man aber sehr selten so viele Einträge benötigt, bleiben meist mehrere Blocks frei.
Das Programm "Blocks use on Track 18" kopiert normale Files nun auf diese freien Blocks. So ist es möglich, mehr als

```
10 rem blocks use on track 18====64
20 rem (p) commodore welt        ==
30 rem ===========================
40 rem (c) by                    ==
50 rem thomas winkler            ==
60 rem                           ==
70 rem version 2.0  40z/ascii    ==
80 rem c 64 + 1541               ==
90 rem ===========================

100 c4$=chr$(017):c2$=chr$(145)
110 c1$=chr$(157):rn$=chr$(018)
120 rf$=chr$(146):c3$=chr$(029)
130 cl$=chr$(147):wh$=chr$(005)
140 dim di(18)
150 dim bl$(17): rem "bl$ = Blocks
lesen
160 dim nd(17,3):rem"neue directory
170 dim dr(17,2) :rem "frei = 1
180 dim ts(17,2) :rem " track secto
r des zu verschiebenden Blocks
190 co=0: xx=0: c1=0: ww=0
200 ge$="a":ge=0
210 ti$="000000"
220 c=56328
230 poke c+7,peek(c+7) and 127
240 poke c+6,peek(c+6) or 128
250 poke c+3,0
260 poke c+2,0
270 poke c+1,0
280 poke c+0,0
290 gosub 2720
300 poke53280,0:poke53281,0:printcl
$wh$
310 gosub 2270
320 print "......Bemerkung :"
330 print "......==========="
340 printc4$c4$"  Bei jeder Eingabe
 kann man das "
350 printc4$"..Inhaltsverzeichnis d
er Diskette mit"
360 printc4$"..";chr$(34);"$";chr$(
34);" einladen."
370 printc4$"..Danach "rn$" Return
"rf$" druecken."
380 poke 198,0:wait 198,1
390 gosub 2270
400 print"..Die zu "rn$" bearbeiten
de "rf$" Diskette"
410 printc4$"  einlegen !!"
420 gosub 2210
430 open 15,8,15,"i0"
440 open 2,8,2,"#0"
450 print#15,"u1 2 0 18 0"
460 print#15,"b-p 2 2"
470 get#2,a$
480 ifa$<>"a"thenprintc4$c4$"..Disk
ette ist schreibgeschuetzt !!"
```

```
490 if a$<>"a" then gosub 2340
500 gosub 2500
510 printc4$"..Validate zur Sicherh
eit (j/n) ?";
520 get a$:if a$="j" or a$=chr$(13)
 then print "j": print#15,"v0": got
o 560
530 if a$="$" then sys 49152
540 if a$<>"n" then 520
550 print "n"
560 gosub 2150
570 printc4$"..Wie viele Blocks sol
l ich besetzen ?"
580 input"....0";an$
590 printc4$c1$c1$c1$
600 ifan$="$"thensys49152:printc2$c
2$c2$:goto580
610 an=val(an$)
620 ifan<0oran>17thenprintc2$c2$c2$
:goto580
630 if an=0 then close 2: close 15:
 end
640 print:print
650 close 2: open 2,8,2,"#0"
660 for i=2 to 18
670 print#15,"b-a 0 18";i
680 input#15,a$,b$,c$,d$
690 printc2$".......................
..............."
700 printc2$"..Block : ";i;" ";a$;"
 ";b$;"..";xx
710 dr(xx,2)=0: dr(xx,0)=18: dr(xx,
1)=i
720 if a$="00" then dr(xx,2)=1:xx=x
x+1
730 if a$="00" then print#15,"b-f 0
 18";i
740 next
750 if an<=xx then 880
760 gosub 2270
770 print "..So viele Blocks stehen
 nicht zur "
780 printc4$"..Verfuegung !! (Nur";
xx;")"
790 printc4$c4$"..Wollen Sie :"
800 printc4$"........1. das Program
m beenden "
810 printc4$"........2. weniger Blo
cks besetzen"
820 printc4$"..oder..3. Programmneu
start ?"
830 geta$:ifa$="1"thenprint#15,"v0"
:goto1950
840 if a$="3" then close 15: run
850 if a$="$" then sys 49152
860 if a$<>"2" then 830
870 poke631,13:poke632,13:poke198,2
:run
880 :
890 gosub 2270
900 a$=rn$" Moment "rf$
910 gosub 2300
920 ab=1
930 print#15,"u1 2 0 18";ab
940 print#15,"b-p 2 1"
950 get#2,a$
960 ns=asc(a$)
970 for qq=0 to 7
980 if co>=an then 1280
990 x=2+(32*qq)
1000 print#15,"b-p 2";x
1010 get#2,a$: a$=a$+chr$(0)
1020 ifasc(a$)=130orasc(a$)=129then
1120
1030 next
1040 ab=ns
1050 if ns<255 then 1100
1060 printc4$c4$"..Nicht genuegend
Blocks besetzt !!
1070 printc4$c4$"..Taste = Programm
neustart"
1080 poke 198,0:wait 198,1
1090 run
1100 print#15,"u1 2 0 18";ns
1110 goto 940
1120 get#2,a$: a$=a$+chr$(0)
1130 if asc(a$)=18 then  1030
1140 ts(co,0)=asc(a$)
1150 get#2,a$: a$=a$+chr$(0)
1160 ts(co,1)=asc(a$)
1170 ts(c1,2)=1
1180 print#15,"u1 2 0 ";ts(co,0);ts
(co,1)
1190 co=co+1:c1=c1+1
1200 print#15,"b-p 2 0"
1210 get#2,a$: a$=a$+chr$(0)
1220 ifasc(a$)=0thenprint#15,"u1 2
0 18";ab:c1=c1+0:goto1030
1230 ts(c1,2)=0
1240 ts(co,0)=asc(a$)
1250 get#2,a$: a$=a$+chr$(0)
1260 ts(co,1)=asc(a$)
1270 if co<an then 1180
1280 printc4$c4$"..Ich habe jetzt s
chon ..."
1290 open 3,8,3,"#1"
1300 printc4$"  0  Block(s) verarbe
itet."
1310 for z=0 to an-1
1320 if ts(z,2)<>0 then nd(ww,0)=ts
(z,0): nd(ww,1)=ts(z,1)
1330 if ts(z,2)<>0 then nd(ww,2)=dr
(z,0): nd(ww,3)=dr(z,1): ww=ww+1
1340 print#15,"u1 2 0";ts(z,0);ts(z
,1)
1350 print#15,"b-p 2 0"
```

```
1360 get#2,a$,b$
1370 nt=asc(a$+chr$(0))
1380 ns=asc(b$+chr$(0))
1390 for o=1 to 254
1400 get#2,a$
1410 if a$="" then a$=a$+chr$(0)
1420 bl$(z)=bl$(z)+a$
1430 next
1440 print#15,"b-p 3 0"
1450 if nt=0 then print#3,chr$(0);c
hr$(ns);:goto 1480
1460 if z=an-1 then print#3,chr$(ts
(an,0));chr$(ts(an,1));:goto 1480
1470 print#3,chr$(dr(z+1,0));chr$(d
r(z+1,1));
1480 for o=1 to 254
1490 print#3,mid$(bl$(z),o,1);
1500 next
1510 print#15,"u2 3 0 18";dr(z,1)
1520 printc2$c3$;z+1
1530 next
1540 close3
1550 gosub 2270
1560 print "..Directory wird neu zu
sammengestellt"
1570 printc4$"..und geschrieben ...
"
1580 ab=1
1590 print#15,"u1 2 0 18";ab
1600 print#15,"b-p 2 0"
1610 get#2,a$,b$
1620 ns=-1
1630 if asc(a$+chr$(0))=18 then ns=
asc(b$)
1640 for i=0 to 7
1650 bp=3+(i*32)
1660 print#15,"b-p 2";bp
1670 get#2,a$,b$
1680 a=asc(a$+(chr$(0)))
1690 b=asc(b$+(chr$(0)))
1700 x=0
1710 if nd(x,0)<>a or nd(x,1)<>b th
en x=x+1:if nd(x,0)=0 and nd(x,1)=0
 then1750
1720 if nd(x,0)<>a or nd(x,1)<>b th
en 1710
1730 print#15,"b-p 2";bp
1740 print#2,chr$(nd(x,2));chr$(nd(
x,3));
1750 next
1760 print#15,"u2 2 0 18";ab
1770 if ns>0 then ab=ns: goto 1590
1780 close2
1790 printc4$c4$"..Ich mache ein 'V
alidate' ..."
1800 print#15,"v0"
1810 input#15,a$,b$,c$,d$
1820 ifa$<>"00"thenprintc4$c4$;a$;"
 ";b$;" ";c$;" ";d$;" !"
1830 ifa$<>"00"then1830
1840 print#15,"i0"
1850 if ge=1 then ge$="b": gosub 24
00
1860 printc4$c4$"..Programm fertig
...":fori=0to5000:next
1870 gosub 2270
1880 print "        Wollen Sie :"
1890 print "        ============"
1900 printc4$c4$"..1. Neue Diskette
"
1910 printc4$"  2. Ende"
1920 get a$:if a$="$" then sys 4915
2
1930 if a$="1" then close 15: run
1940 if a$<>"2" and a$<>chr$(13) th
en 1920
1950 printc4$"Benoetigte 64'er-Zeit
  : ";left$(ti$,2);"h ";mid$(ti$,3,
2);"m ";
1960 print right$(ti$,2);"s"
1970 h=peek(c+3):m=peek(c+2):s=peek
(c+1)
1980 f=1
1990 if h>32 then h=h-128: f=0
2000 h=int(h/16)*10+h-int(h/16)*16:
 on f goto 2030
2010 if h=12 then 2040
2020 h=h+12
2030 if h=12 then h=0
2040 m=int(m/16)*10+m-int(m/16)*16
2050 s=int(s/16)*10+s-int(s/16)*16
2060 h$=str$(h):if len(h$)=2 then h
$=" 0"+right$(h$,1)
2070 m$=str$(m):if len(m$)=2 then m
$=" 0"+right$(m$,1)
2080 s$=str$(s):if len(s$)=2 then s
$=" 0"+right$(s$,1)
2090 ze$=right$(h$,2)+right$(m$,2)+
right$(s$,2)
2100 printc4$"  Benoetigte Gesamt-Z
eit: ";left$(ze$,2);"h ";mid$(ze$,3
,2);"m ";
2110 print right$(ze$,2);"s"
2120 close15:close2
2130 end
2140 :
2150 input#15,a$,b$,c$,d$
2160 if val(a$)=0 then return
2170 close 2: close 15
2180 print a$;" ";b$;" ";c$;" ";d$;
" !"
2190 stop
2200 :
2210 print:printc2$c3$c3$"ok ?...(R
eturn)"
2220 get a$: if a$="$" then sys 491
```

```
52
2230 if a$<>chr$( 13) then 2220
2240 print:printc2$c2$"...........
....."c2$
2250 return
2260 :
2270 printchr$( 147)
2280 print
2290 a$=" Blocks use Tr.18 "
2300 print tab( int( ( 40-len( a$) ) /2) )
a$
2310 printc4$c4$c4$
2320 return
2330 :
2340 printc4$c4$"  Aendern ? ( j/n)"
2350 close 2:close 15
2360 get a$:if a$="j" then 2400
2370 if a$="$" then sys 49152
2380 if a$<>"n" then 2360
2390 end
2400 open15,8,15,"i0": ge=1
2410 open2,8,2,"#0"
2420 print#15,"m-w"chr$( 1)chr$( 1)ch
r$( 1)chr$( 65)
2430 print#15,"u1 2 0 18 0"
2440 print#15,"b-p 2 2"
2450 print#2,ge$
2460 print#15,"u2 2 0 18 0"
2470 print#15,"i0"
2480 return
2490 :
2500 ab=1
2510 di( 0)=1
2520 di( 1)=1
2530 print#15,"u1 2 0 18";ab
2540 print#15,"b-p 2 1"
2550 get#2,a$
2560 if a$="" then a$=a$+chr$( 0)
2570 a=asc( a$)
2580 if a>18 then 2630
2590 if di( a)=0 then di( a)=1: goto
2610
2600 printc4$c4$c4$c4$"  Directory
Fehler !":goto2690
2610 ab=a
2620 goto 2530
2630 for i=0 to 18
2640 if di( i)=1 then print#15,"b-a
0 18";i
2650 next
2660 input#15,a$,b$,c$,d$
2670 if val( a$)<20 or val( a$)=65 th
en return
2680 goto 2700
2690 input#15,a$,b$,c$,d$
2700 printc4$c4$"..";a$;" ";b$;" ";
c$;" ";d$;" !"
2710 goto 2710
2720 restore
2730 for i=49152 to 49406
2740 read x: pokei,x
2750 next
2760 return
2770 :
2780 data 76,192,192,169,36,133,251
,169,251,133,187,169,0,133,188,169,
1,133
2790 data 183,169,8,133,186,169,96,
133,185,32,213,243,165,186,32,180,2
55,165
2800 data 185,32,150,255,169,0,133,
144,160,3,132,251,32,165,255,133,25
2,164
2810 data 144,208,47,32,165,255,164
,144,208,40,164,251,136,208,233,166
,252
2820 data 32,205,189,169,32,32,210,
255,32,165,255,166,144,208,18,170,2
40,6
2830 data 32,210,255,76,79,192,169,
13,32,210,255,160,2,208,198,32,66,2
46,96
2840 data 56,32,240,255,142,0,197,1
40,1,197,160,0,185,0,4,153,0,193,18
5,0,5
2850 data 153,0,194,185,0,6,153,0,1
95,185,0,7,153,0,196,200,192,0,208,
227,96
2860 data 174,0,197,172,1,197,24,32
,240,255,160,0,185,0,193,153,0,4,18
5,0,194
2870 data 153,0,5,185,0,195,153,0,6
,185,0,196,153,0,7,200,192,0,208,22
7,96
2880 data 32,108,192,169,216,160,19
2,32,30,171,32,3,192,32,228,255,201
,13,208
2890 data 249,32,150,192,96,147,17,
17,32,32,32,32,32,32,32,32,32,32,18
,32,194
2900 data 76,79,67,75,83,32,85,83,6
9,32,212,82,46,49,56,32,146,17,17,1
7
2910 data 17,13,0
2920 rem blocks use on track 18====
2930 rem 038911 bytes memory      ==
2940 rem 007673 bytes program     ==
2950 rem 000231 bytes variables   ==
2960 rem 001090 bytes strings     ==
2970 rem 000324 bytes arrays      ==
2980 rem 029593 bytes free        ==
2990 rem ==========================
```

664 Blocks pro Diskette zu belegen.
**Achtung:** Wenden Sie dieses Programm niemals auf Disketten mit „versteckten“ Blocks an, da diese sich möglicherweise auch auf Track 18 aufhalten.

```
10 rem balkengrafik+diagramm====128
20 rem = (p) commodore welt        ==
30 rem ============================
40 rem = (c) by                    ==
50 rem = walter becker             ==
60 rem =                           ==
70 rem = version 7.0  40z/ascii    ==
80 rem = 128 pc + 1541/1571        ==
90 rem ============================
100 color5,2:color4,2:color0,5
110 dima(12)
120 printchr$(147)
130 print".....mittelwert und summe
nberechnung
140 print".....(mit balkengrafik-un
d diagramm)
150 print".....====================
===========":print
160 print" eingabe der werte":print
170 poke21,64
180 input" anzahl der einzugebenden
 werte : ";z:print :print
190 ifz>12thenprinttab(10)chr$(18)"
maximal 12 werte";chr$(146):print:p
rint:goto170
200 input" wert..1 : ";a(1):print
210 input" wert  2 : ";a(2):print
220 input" wert  3 : ";a(3):print
230 input" wert  4 : ";a(4):print
240 input" wert  5 : ";a(5):print
250 input" wert  6 : ";a(6):print
260 input" wert  7 : ";a(7):print
270 input" wert  8 : ";a(8):print
280 input" wert  9 : ";a(9):print
290 input" wert 10 : ";a(10):print
300 input" wert 11 : ";a(11):print
310 input" wert 12 : ";a(12):print:
print
320 print" werte richtig eingegeben
 ? (j/n)
330 getm$:ifm$<>"j"andm$<>"n"then33
0
340 ifm$="j"thenprintchr$(147):goto
360
350 ifm$="n"thenprintchr$(147):goto
160
360 print" mittelwert und summe
370 print"====================":pr
int:print
380 poke21,0:print
390 sm=a(1)+a(2)+a(3)+a(4)+a(5)+a(6
)+a(7)+a(8)+a(9)+a(10)+a(11)+a(12)
400 sm=int(sm*100+0.5)/100
410 mw=sm/z
420 mw=int(mw*100+0.5)/100
430 printusing" summe......= ###,##
#.##";sm :print:print
440 printusing" mittelwert = ###,##
#.##";mw:print:print:print:print
450 print" taste ";chr$(18)"b= balk
engrafik und d= diagramm";chr$(146)
460 print:print
470 printtab(10)"bitte waehlen sie
!":print
480 getm$:ifm$<>"b"andm$<>"d"then48
0
490 ifm$="b"then510
500 ifm$="d"then710
510 graphic1:scnclr
520 char1,12,1,"- balkengrafik -
530 draw,02,170 to 318,170
540 draw,6,25to10,20to14,25to10,20
550 draw,10,20  to 10,170
560 draw,313,166to318,170to313,174t
o318,170
570 box,15,170,25,170-(a(1)),0,1
580 box,40,170,50,170-(a(2)),0,1
590 box,65,170,75,170-(a(3)),0,1
600 box,90,170,100,170-(a(4)),0,1
610 box,115,170,125,170-(a(5)),0,1
620 box,140,170,150,170-(a(6)),0,1
630 box,165,170,175,170-(a(7)),0,1
640 box,190,170,200,170-(a(8)),0,1
650 box,215,170,225,170-(a(9)),0,1
660 box,240,170,250,170-(a(10)),0,1
670 box,265,170,275,170-(a(11)),0,1
680 box,290,170,300,170-(a(12)),0,1
690 gosub910
700 getkeym$:printchr$(147):graphic
0:goto120
710 printchr$(147):graphic1:scnclr
720 char1,12,1," - diagramm -
730 draw,02,170 to 318,170
740 draw,6,25to10,20to14,25to10,20
750 draw,10,20  to 10,170
760 draw,313,166to318,170to313,174t
o318,170
770 draw,20,170-(a(1))to20,170-(a(1
))
780 draw,20,170-(a(1))to45,170-(a(2
))
790 draw,45,170-(a(2))to70,170-(a(3
))
800 draw,70,170-(a(3))to95,170-(a(4
))
810 draw,95,170-(a(4))to120,170-(a(
5))
820 draw,120,170-(a(5))to145,170-(a
(6))
830 draw,145,170-(a(6))to170,170-(a
(7))
840 draw,170,170-(a(7))to195,170-(a
(8))
850 draw,195,170-(a(8))to220,170-(a
(9))
860 draw,220,170-(a(9))to245,170-(a
```

```
(-10))
870 draw,245,170-(a(10))to270,170-(
a(11))
880 draw,270,170-(a(11))to295,170-(
a(12))
890 gosub910
900 getkeym$:printchr$(147):graphic
0:goto120
910 char1,1,22,str$(a(1))
920 char1,4,23,str$(a(2))
930 char1,7,22,str$(a(3))
940 char1,10,23,str$(a(4))
950 char1,13,22,str$(a(5))
960 char1,16,23,str$(a(6))
970 char1,19,22,str$(a(7))
980 char1,23,23,str$(a(8))
990 char1,26,22,str$(a(9))
1000 char1,29,23,str$(a(10))
1010 char1,32,22,str$(a(11))
1020 char1,35,23,str$(a(12))
1030 char1,30,24,"<return>
1040 return
1050 rem balkengrafik+diagramm==128
1060 rem = 122365 bytes memory    =
1070 rem = 003412 bytes programm  =
1080 rem = 000072 bytes variables =
1090 rem = 054697 bytes free (0)  =
1100 rem = 064184 bytes free (1)  =
1110 rem ==========================
```

## BALKENGRAFIK 128

Bei diesem Programm handelt es sich um ein Anwenderprogramm, da es dem Anwender ermöglicht, eine Mittelwert- und Summenberechnung durchzuführen. Veranschaulichung der eingegebenen Werte bietet die Ausgabe einer Balkengrafik und als weitere Wahlmöglichkeit die Diagrammzeichnung auf dem Bildschirm.
Es können 1 bis 12 Werte eingeben werden.
Die Mittelwert und Summenberechnung erfolgt durch das Programm. In der gezeichneten Grafik wird die Werteeingabe ausgeführt, so daß der Werteverlauf ersichtlich ist.
Die verwendeten Variablen A(1) – A(12) sind als Wertevariablen gewählt. Die Anweisungen 'BOX' sind für die Balkengrafik und 'DRAW' für die Koordinaten sowie der Diagrammzeichnung. CHAR bietet die Möglichkeit der Textausgabe auf dem Grafikbildschirm und Einschreibung der Werte in die Bildschirmgrafik.

## FILETRANSFER

Mit diesem Programm ist es möglich, Textdateien zwischen den Programmen Easyscript und Text-O-Mat+ auszutauschen. Leider werden bei diesem Programm die Steuercodes für den Drucker nicht mit berücksichtigt, da dies wahrscheinlich den Rahmen dieses kleinen Programmes sprengen würde. Außerdem fehlen mir in dieser Beziehung detailliertere Unterlagen zu den beiden Programmen. Doch wäre es im Prinzip recht leicht nachzuvollziehen.
Ich habe mich darauf beschränkt, sowohl die Fileart, als auch das RETURN-Zeichen auf den jeweiligen Filetyp anzupassen. Auf weitere Dokumentation verzichte ich, da dies Programm kurz und für den Laien trotzdem noch recht überschaubar ist.

```
10 rem filetransfer ============64
20 rem (p) 12/85 cbm revue team   =
30 rem ===========================
40 rem (c) 3/86 by                =
50 rem autor hermann wellesen     =
60 rem                            =
70 rem version 1.0  40z/ascii     =
80 rem   c-64 + 1541 floppy       =
90 rem ===========================
100 rem
110 poke53280,0:rem rahmenfarbe sch
warz
120 poke53281,0:rem hintergrundfarb
e schwarz
130 poke646,15:rem zeichenfarbe hel
lgrau
140 poke788,52:rem run/stop sperren
150 printchr$(147):rem clear screen
160 printchr$(14):rem kleinschrift
ein
170 input"Dateiname ";n$
180 n1$="x/"+n$
190 print:print"Dateityp"
200 print:print"1) Text-o-mat"
210 print"2) Easyscript"
220 geta$
230 ifa$<"1"ora$>"2"then220
240 ifa$="1"thenn$=left$(n$+"

",15)+"T,p,r":n1$=n1$+",s,w"
250 ifa$="2"thenn1$=left$(n1$+"

",15)+"T,p,w":n$=n$+",s,r"
260 open2,8,2,n$
270 open3,8,3,n1$
280 get#2,z$
290 ifst<>0thenclose2:close3:end
300 z=asc(z$+chr$(0))
310 printz$;
320 ifa$="1"andz=95thenz=13
330 ifa$="2"andz=13thenz=95
340 print#3,chr$(z);
350 goto280
360 rem
370 rem filetransfer ==========64
380 rem 038911 bytes memory    ===
390 rem 001203 bytes program   ===
400 rem 000028 bytes variables ===
410 rem 000000 bytes arrays    ===
420 rem 000072 bytes strings   ===
430 rem 037608 bytes free (0)  ===
440 rem ========================
```

# MIT DEM KOMPRESSOR SPEICHERPLATZ SPAREN

**Basicprogramme mit Basicprogrammen zu verändern ist im Prinzip ganz einfach. Hier wird dies an einem leicht verständlichen Basickompressor gezeigt.**

Jeder, der sich etwas mit Basicprogrammierung beschäftigt, kennt das: Schreibt man ein Programm, so helfen Kommentarzeilen, das Programm übersichtlicher zu gestalten. Auch zusätzliche Leerzeichen machen das Programm leichter lesbar. Leider wird dieser Vorteil mit einem unnötig hohen Speicherplatzverbrauch und einer mit der Länge des Programms verbundenen Verlangsamung erkauft. Daher ist es sinnvoll, das fertige Programm so weit als möglich zu komprimieren. Zuerst wirft man alle REM-Zeilen und die Leerzeichen heraus. Schon allein dadurch kann man ein Programm oft um 30–40 % verkürzen. Eine weitere Verkürzung erreicht man durch Verlängern der Basiczeilen, indem man mehrere Zeilen hintereinanderhängt. Dies kann bis zu einer Basiczeilenlänge von 255 Byte geschehen, wobei hier die Länge der Zeile im Speicher und nicht auf dem Bildschirm gemeint ist. Auf dem Bildschirm kann eine solche Zeile noch länger werden, wenn viele Token wie GOSUB, PRINT vorkommen, die im Speicher ja als je 1 Byte abgelegt sind.

Als letztes bleibt noch eine Umnummerierung der Zeilen. Zum Beispiel braucht GOSUB6 nur 2 Byte, GOSUB60000 jedoch 6 Byte.

Wie all dies in Basic Programm übrigens bis auf kann, soll hier beschrieben werden. Das Programm ist so aufgebaut, daß es möglichst verständlich und übersichtlich ist, ohne dadurch wesentlich an Geschwindigkeit zu verlieren. Ferner können alle Einzelteile mit kleinen Änderungen getrennt in anderen Programmen verwendet werden. Wie kann man nun Basicprogramme verändern? In der Regel beginnt das Programm in der Speicherstelle 2048 (Basicanfang). Beim VC 20 (für den dieses Progrmm übrigens bis auf den Wert von SB in Zeile 32 und den SYS-Befehl in Zeile 134 unverändert übernommen werden

## Zwei Nullen signalisieren das Ende

kann) liegt der Basicanfang in der Grundversion bei 4608, mit einer Erweiterung ebenfalls bei 2048. Dieser Wert ist in den Speicherstellen 45 und 46 enthalten und kann mit PRINT PEEK (45)+256*PEEK(46)–1 herausgelesen werden. In dieser Speicherstelle muß ene 0 stehen (sonst reagiert der Computer auf NEW oder RUN mit SYNTAX ERROR). Dann folgen in 2049 und 2050 der Linkpointer (oder Koppeladresse) der nächsten Zeile und in 2051 und 2052 das LO-Byte und das HI-Byte der Zeilennummer. Danach folgt die eigentliche Basiczeile, welche wieder mit einer 0 abgeschlossen ist. Dann wieder Linkpointer, Zeilennummer ... So geht es weiter, bis das Programm zuende ist. Man erkennt dies an zwei Nullen anstelle eines neuen Linkpointers.

## Endprodukt kleiner als Quellprogramm

Um nun ein Programm durch ein anderes zu verändern, müssen beide im Speicher stehen. Hierzu lädt man das zu verändernde Programm und setzt dann den Basicanfang hoch mit:
PRINT PEEK(46)+1
(Diese Zahl merken:=A)
POKE 44,A:POKE A*256,0:NEW
Jetzt den Kompressor laden und starten. Das Programm fragt zuerst nach der maximalen Bytezahl pro Zeile. Gibt man hier 0 an, so werden nur die Remarks und die Leerstellen entfernt (ohne auf GOTO ... zu prüfen!). Die maximale Zahl ist 255. Dann sind die Zeilen zum Teil nicht mehr veränderbar, wenn sie mehr als 80 Zeichen haben. Will man dies nicht, so kann man 60 angeben. Das weitere macht dann das Programm. Wie es das macht, soll nun im einzelnen für jeden Teil extra erklärt werden: Variablenliste und Subroutinenbibliothek (Zeile 54 – 142). Was die Subroutinen im einzelnen machen, wird durch die REM-Zeilen erklärt. Näheres dazu noch an den Stellen, wo sie gebraucht werden. Hier soll nur noch einmal darauf hingewiesen werden, daß häufig gebrauchte Unterprogramme am Anfang stehen sollten, da ja immer vom Anfang her danach gesucht wird.

REM und BLANK entfernen (Zeile 152 – 186):

Dies ist noch die leichteste Übung. Zuerst muß berechnet werden, von wo bis wo die eigentliche Basiczeile geht (AB, EB in GOSUB 54). Nur innerhalb dieser Werte darf nach dem Code für REM (143) bzw. SPACE (32) gesucht werden. Ferner müssen die Codes innerhalb von Anführungszeichen überlesen werden. Man kopiert also das ganze Programm einmal, wobei es wieder bei 2048 beginnt und man die REM und SPACE einfach ‚übersieht'. LV dient dazu als Laufvariable und zeigt immer auf die Stelle, wo das neue Programm gerade weitergeschrieben wird. Da das Endprodukt kleiner ist als das Quellprogramm, ergeben sich durch das Überschreiben keine Probleme. Natürlich muß die Zeilennummer wieder an die richtige Stelle (NM+3, NM+4) gepoket werden und die Linkadresse neuberechnet (GOSUB 70) und eingepoket werden (NM+1, NM+2). Wird das Programmende gefunden, müssen die entsprechenden Nullen gesetzt werden. Nach der gleichen Methode kann man sich auch andere Möglichkeiten ausdenken, wie zum Beispiel lange Variablennamen auf 2 Stellen kürzen oder Listschutzcodes (SHIFT/L = 204) entfernen. Man muß nur darauf

achten, daß das Programm dadurch nicht länger wird. Wenn es länger werden soll, muß man dies entsprechend berücksichtigen.

AUF SPRUNGADRESSEN UND IF UNTERSUCHEN (Zeile 196 — 232):

Will man zwei Basiczeilen miteinander verketten, so muß vorher geprüft werden, ob die erste kein IF oder GOTO und die zweite nicht durch GOTO, GOSUB oder THEN angesprungen wird. Die entsprechenden Zeilen müssen dann in GS(n) gesperrt werden. Will man diesen Teil ohne den REM-Killer verwenden, muß auch noch auf REM (143) untersucht werden. Kommt hinter einer Zeilennummer nach GOTO oder GOSUB ein Komma, so wird nach einer weiteren Zeilennummer gesucht (ON-Anweisung). Dieser Programmteil könnte zum Beispiel in einer Dokumentationshilfe verwendet werden.

SUPERZEILER
(Zeile 240—278)

Dieser Teil des Programms ist für sich alleine betrachtet eigentlich am interessantesten. Mit ein paar Änderungen kann man ihn unabhängig vom Rest verwenden, um beliebige Basiczeilen miteinander zu verknüpfen. Von den Unterprogrammen werden nur die Zeilen 128—134 und die Zeile 140 gebraucht.
Zuerst die Änderungen:
Folgende Zeilen bitte weglassen:
248, 262—266, 270.
Folgende Zeilen bitte abändern:
246 IF EB<0 THEN GOTO 280
260 IF I=AB AND C=174 THEN LV=LV-4: POKE NM,58: GOTO 274

Superzeiler

```
0 rem********** superzeiler ******
1 rem (c) 1985 rudolf schmid-fabian
2 rem                postfach 105027
3 rem                6900 heidelberg
4 rem*******************************
6 nb=2048:nm=nb
8 lv=nm+4:ab=nb+5:eb=peek(nb+1)+256
*peek(nb+2)-2
10 if eb<0 then goto 100
12 pokenm,0:pokenm+1,1:pokenm+2,1
16 lz=peek(nb+3):pokenm+3,lz
20 hz=peek(nb+4):pokenm+4,hz
24 print"zn="lz+256*hz
28 fori=abtoeb:co=peek(i)
32 if i=ab and co=174 then lv=lv-4:
pokenm,58:goto40
36 lv=lv+1:pokelv,co
40 next
44 nb=eb+1:nm=lv+1
48 goto8
100 pokenm,0:pokenm+1,0:pokenm+2,0
110 ha=int((nm+3)/256):la=(nm+3)-ha
*256
120 print"Sqqqpoke44,8:poke45,";:pr
intla;:print":poke46,";:printha
130 print"qqsys42291:lists"
140 end
```

Folgende Zeilen bitte umnummerieren: von 128—140 nach 280—288.

Nachdem der Superzeiler abgespeichert ist, lädt man das zu verändernde Programm. An den Anfang jeder Zeile, die man mit der vorhergehenden verketten will, schreibt man (in den Zwischenraum nach der Zeilennummer) einen Pfeil nach oben ‚ ↑ '. Dann wird wieder, wie am Anfang beschrieben, der Basicanfang hochgesetzt, der Superzeiler geladen und gestartet. Der allgemeine Aufbau ist ähnlich wie beim REM-Entfernen.

Bemerkenswert ist die Betriebssystemroutine SYS 42291 (beim VC20 SYS 50483), welche die Basiczeilen neu bindet (die Koppeladressen werden neu berechnet). Hierzu müssen natürlich der Basicanfang und das Basicende wieder heruntergesetzt werden. Die entsprechenden Adressen für die Pokebefehle werden errechnet und auf dem Bildschirm ausgedruckt. Nun kommt der „programmierte Direktmodus" zum Einsatz: Die Befehle werden so am Bildschirm ausgeben, daß der Cursor, wenn er wieder sichtbar wird, auf der Befehlszeile steht. In den Tastaturpuffer (631—640) wird der RETURN-Code (13) gepoket und eine der Anzahl der Befehlszeilen entsprechende Zahl in 198 gepoket. Erreicht das Programm nun einen END-Befehl, wird der Tastaturpuffer abgearbeitet und die Befehle ausgeführt.

Wenn Sie den Basickompressor als Ganzes verwenden, fragen Sie sich vielleicht, wie es nun weitergeht. Dies ist in der Tat nicht ganz trivial. Nachdem der Interpreter uns die Arbeit mit dem Berechnen der Koppeladressen abgenommen hat, muß ja wieder auf den oberen Speicherbereich umgeschaltet werden (Zeile 136) Danach steht der Cursor auf ‚GOTO 286:REM RENUMBER' und wartet darauf, daß Sie RETURN drükken, um mit dem Renumber fortzufahren. Wollen Sie die Zeilen nicht neu numeriert haben, so fahren Sie einfach mit dem Cursor auf die erste Poke-Zeile, drücken RETURN und haben nun Ihr komprimiertes Programm.

LAST BUT NOT LEAST — RENUMBER (Zeile 286—340):

Wenn Sie diesen Teil des Kompressors getrennt verwenden wollen, müssen Sie die verwendeten Unterprogramme natürlich mit eingeben. Da es für einen Kompressor nicht sinnvoll ist, die Zeilennummern zu erhöhen, wurde dieser Programmteil so ausgelegt, daß nachher nur kleinere Zeilennummern vorkommen. Sollen die Zeilennummern aber nicht mit 1 beginnen, so kann der Wert für SZ in Zeile 294 entsprechend erhöht werden. Dabei achten Sie bitte darauf, daß die Zeilennummern nicht größer als vorher werden, da das Programm sonst überschrieben wird (größer bedeutet hier mehr Zeichen). Wollen Sie sich jedoch ein ‚perfektes' Renumber selbst programmieren, können auch größere Zeilennummern (als vorher) erreicht werden, indem das Programm zuvor mit einer POKE neu, PEEK(alt)-Schleife oder eleganter mit der Blockverschieberoutine zu höheren Speicherplätzen

hin verschoben wird. Dies muß natürlich von hinten nach vorne geschehen und das Renumberprogramm muß noch weiter hinten im Speicher stehen. (Beispiel: Das Quellprogramm ist 4 KByte lang und soll um 1 KByte verschoben werden. Dann muß der Baiscanfang auf 2048+(4+1)*1024=7168 =28*256 verschoben werden. Dies macht man am besten durch POKE 44, 28: POKE 28*256,0: NEW.)

Vielleicht sind zur Arbeitsweise des Renumber-Programms noch ein paar Anmerkungen interessant. Will man Zeilennummern neu berechnen, so muß man zunächst die Anzahl der Zeilen bestimmen und die Zeilennummern der Reihe nach in NR(n) einsortieren (Zeile 288-292). Danach sucht man im Programm nach GOTO (137), GOSUB (141) und THEN (167) und den darauffolgenden Zeilennummern. Diese müssen dann ersetzt werden (GOSBU 122).

## DER KOMPRESSOR TESTET SICH SELBST:

Nachdem Sie das Programm abgetippt und abgespeichert haben, wollen Sie sich sicher davon überzeugen, ob es auch ohne Fehl und Tadel ist. Dazu schalten Sie sicherheitshalber den Computer kurz aus und wieder ein und laden den Kompressor. Dann setzen Sie den Basicanfang wie oben beschrieben hinter das im Speicher befindliche Programm und laden den Kompressor noch einmal. Starten Sie den Kompressor und geben für die maximale Bytezahl 255 ein und warten ungefähr 4 Minuten. Dann RETURN in der GOTO ... — Zeile drücken und weitere 2 Minuten warten. Jetzt ist der im unteren Speicherbereich befindliche Kompressor komprimiert. Um ihn abspeichern zu können, fahren Sie bitte mit dem Cursor nach oben bis zur 1. Befehlszeile, drücken RETURN und speichern den Kompressor ab. Mit diesem verdichteten Kompressor komprimieren Sie jetzt den unveränderten und vergleichen dann die beiden verdichteten Programme durch VERIFY miteinander. Das hört sich etwas umständlicher an als es ist. Ausprobieren! Sie werden Staunen, um wieviel kleiner das Programm wird.

Übersichtlich geschriebene Programme werden um 40–50 % kürzer. Die oft bei Basickompressoren versprochene Beschleunigung des Programms hängt jedoch stark vom Programmierstil ab. Ein Programm, welches optimal auf Geschwindigkeit geschrieben ist, wird nur um ungefähr 10 % schneller. Aber das ist doch auch schon was. Der Hauptnutzen liegt jedoch darin, daß man Speicherplatz und damit auch Ladezeit einspart.



```
10 rem basickompressor ==========
12 rem (p)      cbm revue team    =
14 rem ==========================
16 rem (c) 03/86 r. schmid-fabian=
18 rem c-64                       =
20 rem ==========================
22 cl$=chr$(147): c4$=chr$(17)
23 p1$=cl$+c4$+c4$+c4$:ho$=chr$(19)
24 p2$=cl$:for i=1to13:p2$=p2$+c4$:
next

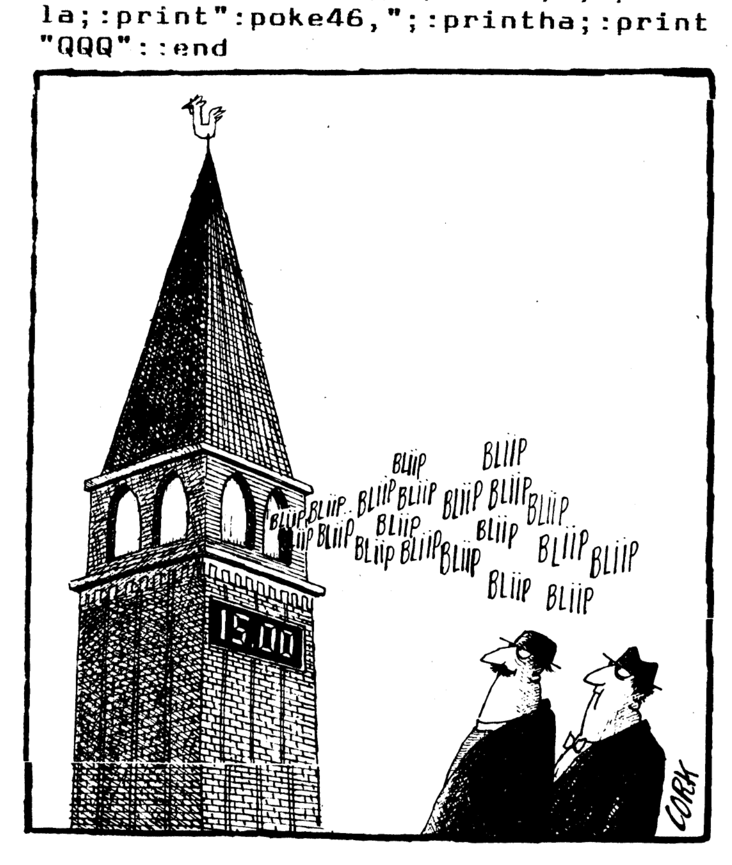
25 rem ***************************
26 rem **** variablenliste ********
27 rem ***************************
28 c=0:rem codezahl im speicher
30 lv=0:lm=0:rem laufvariable zum k
opieren/max. bytes pro zeile
32 nb=2048:nm=nb:rem nulladr. d. ba
siczeile vorh./nachh.
34 zn=0:lz=0:hz=0:rem zeilennummer
36 ab=0:eb=0:rem anfang/ende basicz
eile
38 mz=300:nz=0:lz=0:rem max. zeilen
anzahl/nr.d.zeile/laenge d. zeile
40 dim nr(mz),gs(mz+1):rem zeilennu
mmer/zeilennr. gesperrt
42 goto 142
44 rem ***************************
46 rem  subroutinen bibliothek ***
48 rem ***************************
50 rem basicanfang und -ende *****
52 rem und zeilennummern *********
53 rem ***************************
54 lv=nm+4:ab=nb+5:eb=peek(nb+1)+25
6*peek(nb+2)-2:hk=0
56 if eb<0 then return
58 lz=peek(nb+3):hz=peek(nb+4)
60 zn=lz+256*hz:printho$"zn="zn
62 nz=nz+1:nr(nz)=zn

64 if nz+1>mz then print"mz zu klei
n!":end
66 return
68 rem **neue adresse berechnen***
70 ad=lv+2
72 ha=int(ad/256):la=ad-ha*256
74 pokenm,0:pokenm+1,la:pokenm+2,ha
76 nb=eb+1:nm=lv+1:return
78 rem ** goto,gosub ************
80 z$=""
82 for s=1 to 6:sc=peek(i+s)
84 if sc=44 then 92:rem komma
86 if sc=58 then 92:rem doppelpunkt
88 if sc=0 then 92:rem zeilenende
90 z$=z$+chr$(sc):nexts
92 s=s-1:w=val(z$):gosub114:gs(t)=1
94 if um=1 then return
96 if sc=44 then gs(nz+1)=1:i=i+s+1
```

```
:goto 80
98 return
100 rem  then ********************
102 i=i+1:c=peek(i):if i=eb then re
turn
104 if c=32 then 102
106 if c=137 or c=141 then gosub 80
:return
108 if c>47 then if c<58 then i=i-1
:gosub 80
110 return
112 rem  zeilennr. suchen in nr( )*
114 for t=1 to mz
116 if w=nr(t) then return
118 next:t=0:return
120 rem zeilennr. ersetzen *******
122 z$=str$(t+sz):for ii=2 to len(z
$):a=asc(mid$(z$,ii,1))
124 lv=lv+1:pokelv,a:next
ii:i=i+s:return
126 rem **programm beenden********
128 pokenm,0:pokenm+1,0:pokenm+2,0
130 ha=int((nm+3)/256):la=(nm+3)-ha
*256
132 printp1$"poke44,8:poke45,";:pri
ntla;:print":poke46,";:printha
134 printc4$c4$"sys42291"
136 printc4$c4$"poke43,";:printpeek
(43);:print":poke44,";:printpeek(44
);
138 print":poke45,";:printpeek(45);
:print":poke46,";:printpeek(46):pri
ntho$
140 :poke631,13:poke632,13:poke633,
13:poke198,3:end
142 printcl$:input"maximale bytezah
l pro zeile";lm
144 rem ************************
146 rem rem und blank entfernen
148 rem zeilennummern eintragen
150 rem ************************
152 printcl$c4$"******* erster durc
hlauf ***************"
154 nb=2048:nm=nb:nz=0
156 gosub 54:rem anfang,ende berech
n.
158 pokenm+3,lz:pokenm+4,hz:rem zei
lennr.
160 if eb<0 then pokenm,0:pokenm+1,
0:pokenm+2,0:goto 196
162 for i=ab to eb:c=peek(i)
164 if c=34 then if hk=0 then hk=1:
 goto 178
166 if c=34 then if hk=1 then hk=0:
 goto 178
168 if hk=1 then 178 :rem in anfueh
r.z.
170 if c=32 then 180 :rem blank
172 if c<>143 then 178
174 if peek(i-1)=58 and i>ab then l
v=lv-1:rem doppelpunkt vor rem entf
ernen
176 i=eb: goto 180
178 lv=lv+1:pokelv,c:rem kopieren
180 next i
182 if hk=1 then lv=lv+1:pokelv,34
184 gosub 70:rem neue adresse
186 goto 156
188 rem ************************
190 rem ** auf sprungadr. u. if
192 rem ****** untersuchen ****
194 rem ************************
196 print"****** zweiter durchlauf
***************"
198 if lm=0 then 240
200 nb=2048:nz=0
202 ka=peek(nb+1)+256*peek(nb+2)
204 if ka=0 then 240
206 ab=nb+5:eb=ka-2:nz=nz+1:printho
$"nz="nz
208 for i=ab to eb:c=peek(i)
210 if c=34 then if hk=0 then hk=1:
 goto 228
212 if c=34 then if hk=1 then hk=0:
 goto 228
214 if hk=1 then 228 :rem in anfueh
r.z.
216 if c<137 then 228
218 if c=143 then i=eb: goto 228 :r
em remarks
220 if c=137 then gosub 80:gs(nz+1)
=1:goto 228:rem goto
222 if c=141 then gosub 80:goto 228
:rem gosub
224 if c=139 then gs(nz+1)=1:goto 2
28:rem if
226 if c=167 then gosub 102:rem the
n
228 next i
230 nb=ka-1
232 goto 202
234 rem **********************
236 rem  superzeiler *********
238 rem **********************
240 print"****** dritter durchlauf
**************"
242 nb=2048:nm=nb:nz=0:gs(1)=1
244 lv=nm+4:ab=nb+5:eb=peek(nb+1)+2
56*peek(nb+2)-2:nz=nz+1
246 if eb<0 then printp2$"goto286:r
em renumber ":goto 128
248 if eb<ab then lv=lv-5:sp=1:goto
 276
250 pokenm,0:pokenm+1,1:pokenm+2,1
252 lz=peek(nb+3):pokenm+3,lz
254 hz=peek(nb+4):pokenm+4,hz
```

```
256 printho$"zn="lz+256*hz
258 for i=ab to eb:c=peek(i)
260 if i>ab then 272
262 if sp=1 then sp=0:goto 272
264 if lm=0 then 272
266 if gs(nz)=1 then 272
268 zl=zl+eb-ab+5:if zl>lm then zl=
0:goto272
270 lv=lv-4:pokelv,58
272 lv=lv+1:pokelv,c
274 next
276 nb=eb+1:nm=lv+1
278 goto244
280 rem *********************
282 rem renumber ************
284 rem *********************
286 printcl$c4$"****** vierter durc
hlauf ***************"
288 nb=2048:nm=nb:nz=0:um=1
290 gosub 54:if eb<0 then 294
292 nb=eb+1:nm=nb:goto290
294 nb=2048:nm=nb:nz=0:sz=0:lb=sz+1
:hb=0
296 gosub
54:pokenm+3,lb:pokenm+4,hb:rem anfa
ng,ende berechn.,zn einsetzen
298 lb=lb+1:if lb=256 then lb=0:hb=
hb+1
300 if eb<0 then printcl$:goto 128
302 for i=ab to eb:c=peek(i)
304 if c=34 then if hk=0 then hk=1:
 goto 330
306 if c=34 then if hk=1 then hk=0:
 goto 330
308 if hk=1 then 330 :rem in anfueh
r.z.
310 if c<>137 and c<>141 then 320:r
em goto,gosub
312 lv=lv+1:pokelv,c:gosub 80
314 gosub 122:rem zeilennr. ersetzen
316 if sc=44 then c=44:i=i+1:goto 3
12
318 goto332
320 if c<>167 then 330:rem then
322 lv=lv+1:pokelv,c
324 i=i+1:c=peek(i):if c=32 then 32
4
326 if c>47 then if c<58 then i=i-1
:gosub 80:gosub 122:goto332
328 if c=137 or c=141 then lv=lv+1:
pokelv,c:gosub 80:gosub 122:goto332
330 lv=lv+1:pokelv,c:rem kopieren
332 next i
334 if hk=1 then lv=lv+1:pokelv,34
336 gosub 70:rem neue adresse
338 goto 296
340 fori=2048to2065:printi,peek(i):
next
```

```
0 goto50
1 ******superzeiler**************
2 :der superzeiler kann zeilen bis
zu:::::255 bytes lang machen, dazu
 muss nur:::an den anfang der zeile
, die an die::::vorhergehende ange
haengt werden soll:::ein   gesetzt
 werden. dabei muss man:::darauf a
chten, dass die erste zeile
5 :kein rem oder if enthaelt und di
e::::::zweite nicht durch goto, go
sub, on..:::angesprungen werden sol
l.:::::::::::::::::::c)
1985 rudolf schmid-fabian:::::::::
::::postfach 105027:::::::::::::
:::::::::6900 heidelberg:::::::::
::::
50 nb=2048:nm=nb
60 lv=nm+4:ab=nb+5:eb=peek(nb+1)+25
6*peek(nb+2)-2:if eb<0 then goto 18
0
80 pokenm,0:pokenm+1,1:pokenm+2,1:l
z=peek(nb+3):pokenm+3,lz:hz=peek(nb
+4):pokenm+4,hz:print"zn="lz+256*hz
:fori=abtoeb:co=peek(i):if i=ab and
 co=174 then lv=lv-4:pokenm,58:goto
150
140 lv=lv+1:pokelv,co
150 next:nb=eb+1:nm=lv+1:goto60
180 pokenm,0:pokenm+1,0:pokenm+2,0:
ha=int((nm+3)/256):la=(nm+3)-ha*256
:print"Sqqqpoke44,8:poke45,";:print
la;:print":poke46,";:printha;:print
"QQQ"::end
```



# ANZEIGENSERVICE

**Die große Börse für jeden Zweck in der CBM REVUE / COMMODORE-WELT. Kostenlos für Privat-Inserenten. Spottbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post — Freimachen nicht vergessen! — Unsere Adresse steht auf dem Coupon, ebenso die Preise für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht CBM-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht aufgenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge. Für gewerbliche Anbieter: 5 DM p. mm.**

**CW**
**Anzeigenabteilg.**
**Postfach 1107**
**8044 Lohhof**

Name ____________________
Vorname ____________________
Straße/Hausnr. ____________________
PLZ/Ort ____________________

# DAS SONDERANGEBOT: PRIVATE KLEINANZEIGEN KOSTENLOS!

Das bietet Ihnen COMMODORE-WELT: KLEINANZEIGEN SIND KOSTENLOS FÜR PRIVATANBIETER! Suchen Sie etwas, haben Sie etwas zu verkaufen, zu tauschen, wollen Sie einen Club gründen? Coupon ausfüllen, auf Postkarte kleben oder in Briefumschlag stecken und abschicken. So einfach geht das. Wollen Sie das Heft nicht zerschneiden, können Sie den Coupon auch fotokopieren. Oder einfach den Anzeigentext uns so schicken, auf Postkarte oder im Brief. Aber bitte mit Druckbuchstaben oder in Schreibmaschinenschrift!

Und: Einschließlich Ihrer Adresse und/oder Telefonnummer sollten acht Zeilen à 28 Anschläge nicht überschritten werden.

ACHTUNG: WICHTIGER HINWEIS!
Wir veröffentlichen nur Kleinanzeigen privater Inserenten, keine gewerblichen Anzeigen. Die kosten pro Millimeter DM 5.00 plus Mehrwertsteuer!

Wir versenden für Privat-Inserenten keine Beleg-Exemplare!

Chiffre-Anzeigen sind nicht gestattet! Wir behalten uns vor, Anzeigen, die gegen rechtliche, sittliche oder sonstige Gebote verstoßen, abzulehnen!

Anzeigenabdruck in der Reihenfolge ihres Eingangs, kein Rechtsanspruch auf den Abdruck in der nächsten Ausgabe!

Die Insertion ist nicht vom Kauf des Heftes abhängig!

Wir behalten uns vor, Anzeigen, die nicht zum Themenkreis des Heftes — Computer — gehören, nicht abzudrucken oder sie nur insoweit zu berücksichtigen, wie es der Umfang des kostenlosen Anzeitenteils zuläßt.

# PROGRAMMSERVICE

Hiermit bestelle ich
die Listings dieses Heftes auf
☐ Kassette (10 DM) ☐ Diskette (25 DM)

Zutreffendes bitte ankreuzen!
Ich zahle:
per beigefügtem Scheck / Schein ( )

Gegen Bankabbuchung am Versandtag ( )
Meine Bank (mit Ortsname) ..............................................................

Meine Kontonummer ..............................................................

Meine Bankleitzahl .................... (steht auf jedem Bankauszug)

Vorname ...................... Nachname ..............................................

Str./Nr. ...................... PLZ / Ort ..............................................

Hiermit bestätige ich, Ihre Verkaufsbedingunen gelesen zu haben und zu akzeptieren. (3)

Unterschrift ..............................................................

Bitte ausschneiden und einsenden an
COMMODORE-WELT
KASSETTENSERVICE 5/86
Postfach 1107
8044 Unterschleißheim

---

# LESER WERBEN LESER

***GEWINNEN SIE EINE COMPUTER UHR! Und zusätzlich eventuell noch ein großes Commodore-Buch. Oder ein Paket Disketten. ODER AUCH EINEN COMMODORE-DRUCKER — ODER EINE DISKETTENSTATION! Wie? Sie werben einen Abonnenten. Dann haben Sie auf jeden Fall schon die Computer-Uhr gewonnen. Zusätzlich verlosen wir unter allen, die mitmachen, jeden Monat vier weitere wertvolle Preise. Und alle sechs Monate gibt es einen Hauptpreis unter allen Abo-Werbern zu gewinnen. Also: Mitmachen. Mitgewinnen.***

***Ja, ich mache mit beim Abo-Wettbewerb. Ich habe***




***Herrn/Frau..................***

***..............................***

***Straße/Hausnr. ............***

***..............................***

***PLZ/Ort ......................***

***..............................***

***als neuen Abonnenten der COMMODORE WELT geworben.***

***Der neue Abonnent war bisher noch nicht Bezieher dieser Zeitschrift. Als Prämie erhalte ich nach Eingang des Abo-Entgeltes auf jeden Fall eine Computer-Uhr, wie abgebildet, und nehme zusätzlich noch an der Verlosung des Monats sowie der halbjährlichen Hauptpreise teil. Mir ist bekannt, daß der Rechtsweg bei den Verlosungen ausgeschlossen ist. Meinen Preis senden Sie an***

***Name .........................***

***Straße/Hsnr. ..............***

***PLZ/Ort ......................***

***(Bitte ausschneiden und zusammen mit der Abobestellkarte unten einsenden!)***




**Rund 150 Seiten Listings TIPS + TRICKS**

**Anwender-Programme**
**Utilities**
**Schul-Programme**
**Spiele**
**Adventures**

Rund 150 Seiten, voll mit Listings, Service, Tips und Tricks. Ein Muß für jeden Commodore-User, ob mit dem VC 20 oder dem 128 — selbstverständlich erst recht für den 64er Fan!

# RESERVIERUNGS-KARTE

✂- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

Bitte senden Sie mir (   ) Exemplar(e)
COMMODORE-WELT SPECIAL Nr. 1/86
zum Preis von DM 14,80

Ich zahle
(   ) Per beigefügtem Scheck*
(   ) Per Bankabbuchung am Versandtag*

Name ______________________________

Straße/Hausnr. ______________________________

Ort/PLZ ______________________________

Bei Bankabbuchung:

Kto-Nr.: ______________ Bankleitzahl ______________

Bei (Bank und Ort) ______________________________

Unterschrift - - - - - - - - - - - - - - - - - - - -

Ausschneiden und einsenden

* = Abbuchung erst am Versandtag!

**COMMODORE WELT Reservierungs-Service**
**Postfach 1107**
**8044 Lohhof**




Das unabhängige Commodore-Magazin

# KOMMT REGELMÄSSIG ZU IHNEN INS HAUS

Finden Sie Ihre COMMODORE-WELT nicht am Kiosk? Weil sie schon ausverkauft ist? Oder „Ihr" Kiosk nicht beliefert wurde? Kein Problem! Für ganze 30 DM liefern wir Ihnen per Post sechs Hefte ins Haus (Ausland 40 DM). Einfach den Bestellschein ausschneiden – fotokopieren oder abschreiben, in einen Briefumschlag und ab per Post (Achtung: Porto nicht vergessen). COMMODORE-WELT kommt dann pünktlich ins Haus.



**WICHTIGE RECHTLICHE GARANTIE!**
Sie können diesen Abo-Auftrag binnen einer Woche nach Eingang der Abo-Bestätigung durch den Verlag widerrufen – Postkarte genügt. Ansonsten läuft dieser Auftrag jeweils für sechs Ausgaben, wenn ihm nicht vier Wochen vor Ablauf widersprochen wird, weiter.

# NUTZEN SIE UNSEREN BEQUEMEN POSTSERVICE

# ABO SERVICE-KARTE

5/86

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

**COMMODORE WELT**

Abo-Servie 5/86
Postfach 1107
8044 UNTERSCHLEISSHEIM

**Coupon**

**Ja, ich möchte von Ihrem Angebot Gebrauch machen.**
Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten sechs Ausgaben an untenstehende Anschrift. Wenn ich nicht vier Wochen vor Ablauf kündige, läuft diese Abmachung automatisch weiter.

Name ______________________ (5)
Vorname ______________________
Straße/Hausnr. ______________________
Plz/Ort ______________________
Ich bezahle:
☐ per beiliegendem Verrechnungsscheck
☐ gegen Rechnung
☐ bargeldlos per Bankeinzug von meinem Konto
bei (Bank) und Ort ______________________
Kontonummer ______________________
Bankleitzahl ______________________
(steht auf jedem Kontoauszug)
Unterschrift ______________________

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

Unterschrift ______________________

# MIXED 16/116

```
10 rem mixed datei =============c16
20 rem (p) 04/86 commodore-welt bw=
30 rem =============================
40 rem (c) 01/86 by                =
50 rem harald beiler               =
60 rem                             =
70 rem version 3.S 40z/ascii       =
80 rem c16/p4 + 1530/1541/drucker  =
90 rem =============================
```

Als erstes erfolgt die Auswahl zwischen einer reinen Textdatei oder einer gemischten Zahlen/Textdatei.

Nach Erscheinen des Menüs können folgende Unterprogramme aufgerufen werden:
Neue Datei eingeben
Speichern einer Datei
Bestehende Datei laden
Anfügen an bestehende Datei
Daten auf Bildschirm
Print (Daten auf Drucker)
Löschen der Daten
Umschalten zur Zahlen-Text-Datei
Ende

Nach Aufruf des Modus >Neueingabe< erfolgt die Abfrage >Dateiname<. Nach Ende Daten speichern, falls nötig, Datendisk einlegen.

Menüpunkt >Speichern einer Datei<: Sollte es Probleme beim Abspeichern der Datei geben, wird automatisch eine Disk-Fehlermeldung in Klartext gedruckt. Menüpunkt >Daten laden<: Nach Abfrage des Dateinamens erfolgt entweder eine Übernahmebestätigung oder eine Disk-Fehlermeldung in Klartext.

Bei der allgemeinen Textdatei sind Komma, Semikolon, Doppelpunkt, Anführungsstriche zur Eingabe erlaubt.

Menüpunkt >Anfügen an eine bestehende Datei<: Sind keine Daten im Speicher, informiert der Computer darüber. Es wird bei bestehenden Daten die Anzahl angezeigt, um weitere Daten anzufügen. Eingabe mit >*< beenden. Danach Abfrage >Daten speichern?<, falls nötig, Datendisk einlegen.

Menüpunkt >Daten auf Bildschirm<: Abfrage >Einzelschritte per Tastendruck oder angezeigt, Eingabe beenden mit >*<

Menüpunkt>Drucken<: Nach Aufruf dieses Programmteils ist zur Druckeranpassung die Änderung der Sekundäradresse möglich. Drucker einschalten nicht vergessen, sonst Programmabsturz!

Menüpunkt >Löschen der Daten<: Nach einer Sicherheitsabfrage sind dann die Daten unwiderbringlich futsch und der Arbeitsspeicher geleert.

Menüpunkt >Umschalten zur Zahlen und Textdatei< (Oder umgekehrt): Nach Anzeigen des Zwischenmenüs Text- und Zahlendatei sind folgende Auswahlpunkte möglich:
Neueingabe
File laden
File speichern
File anzeigen
File addieren
File erweitern
Druckerroutine
Ende
Umschalten auf Textdatei

```
100 color 0,1:color4,1:color1,14:sc
nclr:printchr$(14)chr$(11)chr$(27)c
hr$(69)
110 char1,4,5,"T = reine Textdatei.
..........",1
120 char1,4,7,"Z = Text mit Zahlen
+ Addition",1
130 char1,0,10,"Bitte druecken Sie
entsprechende Taste!":print
140 getkeya$
150 ifa$<>"t"anda$<>"z"then140
160 ifa$="t"thengosub180:goto240
170 ifa$="z"thengosub180:goto1890
180 lz$=".........................:
.............":rem 39 leerzeichen
190 dz$="###,###.##"
200 ze=1000:dima$(ze):n=1000:dimn(n
):return
210 rem ************************
220 rem *************** titel *
230 rem ************************
240 gosub1830:char1,8,3,"Allgemeine
 Text Datei",1:print:gosub660
250 char1,2,6,"Eingabe: Dateiname o
der Nummer":print:char1,11,7,"die
Daten":print:gosub660
260 char1,2,10,"(Alle Eingaben auss
er Anfuehrungs-":print
270 char1,2,11,"zeichen sind erlaub
t !)":print:gosub660
280 char1,12,18,"M = M E N U E",1:p
rint
290 getkeya$:ifa$="m"then300:else:g
oto290
300 scnclr:gosub690:gosub1130
310 getkeyq$
320 ifq$="n"goto450
330 ifq$="h"goto1230
340 ifq$="b"goto900
350 ifq$="d"goto960
360 ifq$="a"goto1560
370 ifq$="e"thenend
380 ifq$="l"goto1750
390 ifq$="s"goto800
400 ifq$="u"goto1890
410 goto310
420 rem ************************
```

```
430 rem *********** neueingabe *
440 rem ************************
450 scnclr
460 char1,15,2,"NEUEINGABE",1:print
:gosub660:gosub680
470 printchr$( 17)tab(2)"Dateiname:
";:poke19,4:inputcm$:poke19,0
480 gosub490:goto730
490 print:ze=1:
500 printtab(2);:poke1319,34:poke23
9,1:inputa$(ze):printchr$( 13);:tl=z
e/15:fori=1to40
510 iftl=ithengosub580:printchr$( 19
)"Seite: "tl+1:char1,2,8,""
520 next
530 ifa$(ze)=chr$( 42)thenreturn
540 ze=ze+1:poke239,0
550 goto500
560 rem ************************
570 rem ******* unterprogramme *
580 rem ************************
590 printchr$( 18);:char1,2,22,"Seit
e vor=↑":char1,20,22,"Menue =£ "
600 getkeya$
610 ifa$=chr$( 145)then640
620 ifa$=chr$( 92)then300
630 getkeya$
640 char1,0,8,"",0:i=0: dountili=15
:printlz$:i=i+1:loop
650 char1,0,8,"":return
660 printtab( 1):i=0:dountili=36:pri
ntchr$( 96);:i=i+1:loop:print
670 return
680 char1,7,5,"(Eingabe beeenden mi
t '*')":print:return
690 char1,15,2,"M E N U E",1:print:
gosub660:return
700 i=1:dountili=ze
710 ifa$(i)=""then1700:loop
720 return
730 char1,8,22,"Daten speichern ? (
j/n)",1:print
740 printtab(2)chr$( 18)"Falls noeti
g, Datendisk einlegen !"chr$( 146)
750 getkeya$
760 ifa$="j"then800:else:goto300
770 rem ************************
780 rem ***** daten speichern *
790 rem ************************
800 scnclr:char1,0,2,"Daten speiche
rn",1:print
810 print"File-Name oder -nummer: "
;cm$
820 print:printcm$" wird jetzt gesi
chert":print"Bitte warten.":print
830 open2,8,2,"@0:"+cm$+",s,w"
840 print#2,ze
850 i=1:dountili=ze:print#2,chr$( 34
)+a$(i)+chr$( 34):i=i+1:loop:close2
860 gosub1650:goto1030
870 rem ************************
880 rem ********* daten laden *
890 rem ************************
900 scnclr:char1,0,2,"Daten laden."
,1:print
910 print"Dateiname: ";:poke19,4:in
putcm$:poke19,0:print
920 open2,8,2,"@0:"+cm$+",s,r":prin
tcm$
930 input#2,ze
940 i=1:dountili=ze:input#2,a$(i):i
=i+1:loop:close2
950 gosub1650:goto300
960 scnclr:char1,0,2,"Bildschirm-Au
sgabe",1:print
970 printtab(2)chr$( 18)" E "chr$( 14
6)"inzelschritt per Tastendruck"
980 printtab(2)chr$( 18)" G "chr$( 14
6)"esamte Datei anzeigen":gosub660
990 getkeya$
1000 ifa$="e"then1360
1010 ifa$="g"then1450
1020 else:goto990
1030 printtab(2)chr$( 17)chr$( 18)" D
 = Drucken "tab(26)" M = Menue"chr$
( 146)
1040 getq$
1050 ifq$="d"thengosub1230:goto300
1060 ifq$="m"then300
1070 goto1040
1080 goto1040
1090 else:goto1040
1100 rem ************************
1110 rem **** ein/ausgabemenue *
1120 rem ************************
1130 printchr$( 17)tab(2)chr$( 18)" N
 "chr$( 146)"eue Datei eingeben":pri
nt
1140 printtab(2)chr$( 18)" S "chr$( 1
46)"peichern einer Datei":print
1150 printtab(2)chr$( 18)" B "chr$( 1
46)"estehende Datei laden":print
1160 printtab(2)chr$( 18)" A "chr$( 1
46)"nfuegen an bestehende Datei":pr
int
1170 printtab(2)chr$( 18)" D "chr$( 1
46)"aten auf Bildschirm":print
1180 printtab(2)chr$( 18)" H "chr$( 1
46)"ardcopy (Druckerausgabe)":print
1190 printtab(2)chr$( 18)" L "chr$( 1
46)"oeschen von Daten":print
1200 printtab(2)chr$( 18)" U "chr$( 1
46)"mschalten zur Zahlen/Textdatei"
:gosub660
1210 printtab(2)chr$( 18)" E "chr$( 1
46)"NDE":gosub660:return
```

```
1220 rem ***********************
1230 rem **** daten ausdrucken *
1240 rem ***********************
1250 scnclr:char1,0,2,"Drucker-Rout
ine.",1:print
1260 print:print" Aenderung der "
1270 print" Sekundaeradresse"
1280 printtab(19)chr$(145)"7"chr$(1
57)chr$(157)chr$(157);:inputsa
1290 printchr$(17)chr$(18)"Wenn Dru
cker bereit ist, Taste !"chr$(146)
1300 poke239,0:wait239,1
1310 open4,4,sa:cmd4
1320 print#4,cm$:print#4:i=1:dounti
li=ze:print#4,a$(i):i=i+1:loop:prin
t#4:close4:goto300
1330 rem ***********************
1340 rem ** bildschirm-ausgabe *
1350 rem ***********************
1360 gosub700
1370 printcm$:print:ze=1
1380 printtab(2)a$(ze):tl=ze/15:pok
e239,0:wait239,1
1390 forj=1to40
1400 iftl=jthengosub640
1410 next
1420 ifa$(ze)=chr$(42)then1030
1430 ze=ze+1
1440 goto1380
1450 gosub700
1460 printcm$:print:ze=1
1470 printtab(2)a$(ze):tl=ze/15:for
j=1to40
1480 iftl=jthenprintchr$(18)"Seite
vor=Taste."chr$(146):poke239,0:wait
239,1:gosub640
1490 next
1500 ifa$(ze)=chr$(42)then1030
1510 ze=ze+1
1520 goto1470
1530 rem ***********************
1540 rem append-daten anhaengen*
1550 rem ***********************
1560 scnclr:gosub700:gosub1620:ze=z
e-1
1570 gosub680:gosub660:printtab(2)c
m$:print:printtab(2)a$(ze)
1580 ze=ze+1:printtab(2);:poke1319,
34:poke239,1:poke19,4:inputa$(ze):p
oke19,0:print
1590 ifa$(ze)=chr$(42)then1610
1600 goto1580
1610 print:goto730
1620 ifze=900then1700
1630 printtab(2)chr$(18)"Datensaetz
e bisher:"str$(ze-1)chr$(146)
1640 printtab(2)" Letzter Datensatz
 wird angezeigt.":return
1650 rem ***********************
1660 rem ********* disk-fehler *
1670 rem ***********************
1680 ifds<>0thenprintds$;ds:cm$="":
goto300
1690 return
1700 gosub1830:printchr$(19)chr$(17
)tab(3)chr$(18)" Keine Daten im Spe
icher !!!"chr$(146)
1710 char1,25,18,"M = MENUE",1:prin
t:goto290
1720 rem ***********************
1730 rem ****** daten loeschen *
1740 rem ***********************
1750 gosub1830:print:printchr$(19)t
ab(2)chr$(17)chr$(18)"Daten loesche
n."chr$(146)
1760 print:printtab(2)"Sind Sie sic
her (j/n) ?":gosub660
1770 getkeya$
1780 ifa$="j"then1790:else:goto300
1790 printtab(2)"Bitte warten...":i
=1:dountili=ze:a$(i)="":i=i+1:loop:
cm$=""
1800 print:printtab(2)chr$(18)" Arb
eitsspeicher ist jetzt leer."chr$(1
46)
1810 fort=1to200:next:goto300
1820 char1,0,10,"":i=1:dountili=10:
printlz$;:loop:return
1830 scnclr
1840 printchr$(176);:i=0
1850 dountili=36:printchr$(96);:i=i
+1:loop:printchr$(174):j=0
1860 dountilj=20:printchr$(125)tab(
37)chr$(125):j=j+1:loop:k=0
1870 printchr$(173);:dountilk=36:pr
intchr$(96);:k=k+1:loop:printchr$(1
89)
1880 return
1890 gosub1830:gosub690
1900 print:printtab(10)chr$(18)"TEX
T/ZAHLENDATEI"chr$(146):print
1910 printtab(9)" 1 = Neueingabe"
1920 printtab(9)" 2 = File laden"
1930 printtab(9)" 3 = File speicher
n"
1940 printtab(9)" 4 = File anzeigen
"
1950 printtab(9)" 5 = File addieren
"
1960 printtab(9)" 6 = File erweiter
n"
1970 printtab(9)" 7 = Drucker-Routi
ne"
1980 printtab(9)" 8 = ENDE
1990 print:printtab(9)" 9 = Umschal
ten auf "
```

```
2000 printtab(14)"Textdatei"
2010 getq$:ifq$=""then2010
2020 onval(q$)goto2040,2190,2250,23
30,2430,2580,2700,2860,300
2030 goto2010
2040 rem ************************
2050 rem ****** neueingabe t/z *
2060 rem ************************
2070 pn=len("Neueingabe eines Files
")
2080 scnclr:char1,(40-pn)/2,2,"Neue
ingabe eines Files",1:print:gosub66
0
2090 gosub680:gosub660
2100 poke19,4:input"File Name: ";fl
$:poke19,0:print:ifn=1000thenn=0
2110 print"EG-Text...";:inputsb$(n)
:print
2120 ifsb$(n)=chr$(42)then1890
2130 print"EG-Zahl...";:inputz(n):p
rint
2140 n=n+1
2150 goto2110
2160 rem ************************
2170 rem ********** file laden *
2180 rem ************************
2190 scnclr:printchr$(18)"File lade
n."chr$(146)
2200 poke19,4:input"File-Name: ";fl
$:poke19,0:print
2210 open2,8,2,fl$+",s,r":input#2,n
2220 fori=0ton:input#2,sb$(i):input
#2,z(i):next:close2
2230 goto1890
2240 rem ************************
2250 rem ****** file speichern *
2260 rem ************************
2270 scnclr:printfl$" wird gesicher
t. Bitte warten..."
2280 open2,8,2,"@0:"+fl$+",s,w"
2290 print#2,n
2300 i=0:dountili=n:print#2,sb$(i):
print#2,z(i):i=i+1:loop:close2
2310 goto1890
2320 rem ************************
2330 rem ******* file anzeigen *
2340 rem ************************
2350 scnclr:printchr$(18)"File anze
igen."chr$(146)
2360 print"File-Name: "fl$
2370 iffl$=""then2680
2380 i=0:dountili=n
2390 printsb$(i)tab(25);:print usin
g dz$;z(i):i=i+1:loop
2400 gosub2870
2410 goto1890
2420 rem ************************
2430 rem ************ addition *
2440 rem ************************
2450 scnclr:printchr$(18)"Werte add
ieren."chr$(146)
2460 print"File-Name: "fl$
2470 iffl$=""then2680
2480 s=0:i=0:dountili=n
2490 printsb$(i)tab(25);:print usin
g dz$;z(i):s=s+z(i):i=i+1:loop
2500 printtab(27)"---------":printta
b(25);:print using dz$;s
2510 print:print:printtab(10)chr$(1
8)"Ausdrucken j/n ?"chr$(146)
2520 getx$:ifx$="n"then1890
2530 ifx$="j"then2700
2540 goto2520
2550 gosub2870
2560 goto1890
2570 rem ************************
2580 rem ************** append *
2590 rem ************************
2600 scnclr:printchr$(18)"File erwe
itern/Daten anhaengen"chr$(146)
2610 print"File-Name: "fl$
2620 iffl$=""then2680
2630 printsb$(n-1),z(n-1)
2640 :input"EG-Text...:";sb$(n):pri
nt:ifsb$(n)=chr$(42)then1890
2650 input"EG-Zahl...:";z(n):print
2660 n=n+1
2670 goto2640
2680 char1,7,12,"Keine Daten im Spe
icher!",1:forq=1to300:nextq:goto189
0
2690 rem ************************
2700 rem ****** druckerroutine *
2710 rem ************************
2720 scnclr:printchr$(18)"Daten dru
cken."chr$(146)
2730 print"Aenderung der"
2740 print"Sekundaeradresse:"
2750 printtab(19)chr$(145)"7"chr$(1
57)chr$(157)chr$(157);
2760 inputsa
2770 print:printchr$(18)"Wenn Druck
er bereit ist, Taste druecken."chr$
(146)
2780 poke239,0:wait239,1
2790 open4,4,sa:cmd4
2800 print#4,tab(5)fl$
2810 fori=0ton-1
2820 print#4,tab(5)sb$(i)tab(40-len
(sb$(i))-len(dz$));:print#4,using d
z$;z(i):s=s+z(i):next
2830 print#4,tab(37)"---------"
2840 print#4,tab(45-len(dz$));:prin
t#4,using dz$;s
2850 print#4:close4:goto1890
2860 scnclr:printchr$(12)chr$(142):
```

```
end
2870 printchr$( 17)chr$( 17)chr$( 18)"
Taste fuer 'MENUE' "chr$( 146)
2880 poke239,0:wait239,1
2890 return
2900 rem mixed datei ===========c16
2910 rem 060671 bytes memory    ===
2920 rem 008731 bytes program   ===
2930 rem 000000 bytes variables ===
2940 rem 000000 bytes arrays    ===
2950 rem 000000 bytes strings   ===
2960 rem 051625 bytes free (0)  ===
2970 rem ==========================
```

# FRANZÖSISCHE VERBEN

Das Programm ist aufgebaut in ein 3er System. Zuerst springt der Rechner zum Anfangsbild. Dann zur Anleitung und schließlich zur Eingabe. So funktioniert auch das zweite Programm.
Bei dem französischem Programm braucht man nicht auf die Besonderheiten dieser Sprache aufzupassen. Circonflex und andere Besonderheiten gibt es in dem Programm nicht.
Bei der Eingabe kann man es normal eingeben. etre, connaitre, ecrire, eteindre, usw.
Insgesamt beinhaltet das Programm die wichtigsten Verben (68 Stück).

**Wichtig:**
Das Programm konjugiert die Verben im Präsens durch. Z.B.:
Verb – etre:
Singular
suis
est
es
Plural
sommes
etes
sont

```
10 rem französische verben   =====64
20 rem (p) commodore welt        ==
30 rem ============================
40 rem (c) by                    ==
50 rem stefan draheim            ==
60 rem                           ==
70 rem version 2.0  40z/ascii    ==
80 rem c 64  +  1530/1541        ==
90 rem ============================

100 goto980
110 rem ***eingabe***
120 s=69:dimb$(s,7)
130 forz=1tos:fory=1to7:readb$(z,y)
:nexty:nextz
140 printchr$( 147)
150 poke53280,0:poke53281,0
160 input"bitte geben sie das gewue
nschte verb ein";i$
170 forz=1tos:ifi$=b$(z,1)thengoto2
00
180 nextz
190 print:print"verb nicht vorhande
n":fort=1to1500:next:goto140
200 printchr$( 147)
210 poke53280,0:poke53281,0
220 print"das verb ist:":printb$(z,
1)
230 print:print"der singular ist:":
fory=2to4:printb$(z,y);"..":next
240 print:print"der plural ist:":fo
ry=5to7:printb$(z,y);"..":next
250 print:input"noch einmal <j/n>";
a$
260 ifa$="j"thengoto140
270 ifa$="n"then sys 64738
280 ifa$<>"j"ora$<>"n"thenprint"fal
sche eingabe":fort=1to999:next:goto
250
290 data aller,vais,vas,va,allons,a
llez,vont
300 data abattre,abats,abats,abat,a
battons,abattez,abattent
310 dataaccueillir,accueille,accuei
lles,accueille,accueillons,accueill
ez,accueillent
320 data apercevoir,apercois,aperco
is,apercoit,apercevons,apercevez,ap
ercoivent
330 dataappartenir,appartiens,-tien
s,appartient,appartenons,appartenez
,appartiennent
340 datas'asseoir,m'assieds,t'assie
ds,s'assied,ns asseyons,vs asseyez,
s'asseyent
350 dataavoir,ai,as,a,avons,avez,on
t
360 databattre,bats,bats,bat,batton
s,battez,battent
370 databoire,bois,bois,boit,buvons
,buvez,boivent
380 databouillir,bous,bous,bout,bou
tons,boutez,boutent
390 datacombattre,combats,combats,c
ombat,combattons,combattez,combatte
nt
400 dataconclure,conclus,conclus,co
nclut,concluons,concluez,concluent
410 dataconduire,conduis,conduis,co
nduis,conduisons,conduisez,conduise
nt
420 dataconnaitre,connais,connais,c
onnait,connaissons,connaissez,conna
issent
430 dataconstruire,construis,-strui
s,construit,constuisons,construisez
,construisent
440 datacontenir,contiens,contiens,
contient,contenons,contenez,contien
nent
```

```
450 datacommettre,commets,commets,c
ommet,commettons,commettez,commette
nt
460 datacoudre,couds,couds,coud,cou
sons,cousez,cousent
470 datacourir,cours,cours,court,co
urons,courez,courent
480 datacouvrir,couvre,couvres,couv
re,couvrons,couvrez,couvrent
490 datacraindre,crains,crains,crai
nt,craignons,craignez,craignent
500 datacroire,crois,crois,croit,cr
oyons,croyez,croient
510 datacuire,cuis,cuis,cuit,cuison
s,cuisez,cuisent
520 datadecrire,decris,decris,decri
t,decrivons,decrivez,decrivent
530 datadetruire,detruis,detruis,de
truit,detruisons,detruisez,detruise
nt
540 datadevoir,dois,dois,doit,devon
s,devez,doivent
550 datadire,dis,dis,dit,disons,dit
es,disent
560 datadormir,dors,dors,dort,dormo
ns,dormez,dorment
570 dataecrire,ecris,ecris,ecrit,ec
rivons,ecrivez,ecrivent
580 dataelire,elis,elis,elit,elison
s,elisez,elisent
590 dataenvoyer,envoie,envoies,envo
ie,envoyons,envoyez,envoient
600 dataeteindre,eteins,eteins,etei
nt,eteignons,eteignez,eteignent
610 dataetre,suis,est,es,sommes,ete
s,sont
620 dataexclure,exclus,exclus,exclu
t,excluons,exluez,excluent
630 datafaire,fais,fais,fait,faison
s,faites,font
640 datafuir,fuis,fuis,fuit,fuyons,
fuyez,fuient
650 datajoindre,joins,joins,joint,j
oignons,joignez,joignent
660 datalire,lis,lis,lit,lisons,lis
ez,lisent
670 datamentir,mens,mens,ment,mento
ns,mentez,mentent
680 datamettre,mets,mets,met,metton
s,mettez,mettent
690 datamourir,meurs,meurs,meurt,mo
urons,mourez,mourent
700 datanaitre,nais,nais,nait,naiss
ons,naissez,naissent
710 dataoffrir,offre,offres,offre,o
ffrons,offrez,offrent
720 dataouvrir,ouvre,ouvres,ouvre,o
uvrons,ouvrez,ouvrent
730 dataparaitre,parais,parais,para
it,paraissons,paraissez,paraissent
740 datapartir,pars,pars,part,parto
ns,partez,partent
750 datapeindre,peins,peins,peint,p
eignons,peignez,peignent
760 dataplaindre,plains,plains,plai
nt,plaignons,plaignez,plaignent
770 dataplaire,plais,plais,plait,pl
aisons,plaisez,plaisent
780 datapleuvoir,pleus,pleus,pleut,
pleuvons,pleuvez,pleuvent
790 datapouvoir,peux,peux,peut,pouv
ons,pouvez,peuvent
800 dataprendre,prends,prends,prend
,prenons,prenez,prennent
810 dataproduire,produis,produis,pr
oduit,produisons,produisez,produise
nt
820 datarecevoir,recois,recois,reco
it,recevons,recevez,recoivent
830 datarire,ris,ris,rit,rions,riez
,rient
840 datasavoir,sais,sais,sait,savon
s,savez,savent
850 datasentir,sens,sens,sent,sento
ns,sentez,sentent
860 dataservir,sers,sers,sert,servo
ns,servez,servent
870 datasortir,sors,sors,sort,sorto
ns,sortez,sortent
880 datasouffrir,souffre,souffres,s
ouffre,souffrons,souffrez,souffrent
890 datasuffire,suffis,suffis,suffi
t,suffisons,suffisez,suffisent
900 datasuivre,suis,suis,suit,suivo
ns,suivez,suivent
910 datatenir,tiens,tiens,tient,ten
ons,tenez,tiennent
920 datavaloir,vaux,vaux,vaut,valon
s,valez,valent
930 datavaincre,vaincs,vaincs,vainc
,vainquons,vainquez,vainquent
940 datavenir,viens,viens,vient,ven
ons,venez,viennent
950 datavivre,vis,vis,vit,vivons,vi
vez,vivent
960 datavoir,vois,vois,voit,voyons,
voyez,voient
970 datavouloir,veux,veux,veut,voul
ons,voulez,veulent
980 rem ***anfangsbild***
990 printchr$(147)
1000 poke53280,0:poke53281,0
1010 printtab(8)"unregelmaessige ve
rben"
1020 print:printtab(12)"franzoesisc
h"
```

```
1030 print:print:print
1040 printtab(8)"(c) copyright by s
.d."
1050 print:print:print:print:print
1060 printtab(8)"bitte taste drueck
en"
1070 geta$:ifa$=""then1070
1080 goto1090
1090 rem ***anleitung***
1100 printchr$(147)
1110 print:printtab(10)"a n l e i t
 u n g"
1120 print:print"das programm beher
rscht die konjungation"
1130 print
1140 print"der unregelmaessigen ver
ben im praesens."
1150 print
1160 print"sie brauchen nur den frz
.infinitiv "
1170 print:print
1180 print"eingeben und schon konju
giert das"
1190 print:print
1200 print"programm alle personen i
m praesens durch"
1210 print
1220 printtab(8)"bitte taste drueck
en"
1230 geta$:ifa$=""then1230
1240 printchr$(147)
1250 print"sie brauchen nicht die f
rz.buchstaben"
1260 print:print"wie circonflex ode
r andere zu beachten."
1270 print:print"sie koennen es nor
mal hinschreiben!!"
1280 print:print"z.b.:"
1290 print:print"etre oder connaitr
e."
1300 print"-.............-"
1310 print:printtab(8)"bitte taste
druecken"
1320 geta$:ifa$=""then1320
1330 goto110

1340 rem franzoesische verben====64
1350 rem 038911 bytes memory     ==
1360 rem 005910 bytes program    ==
1370 rem 000035 bytes variables  ==
1380 rem 000008 bytes strings    ==
1390 rem 001689 bytes arrays     ==
1400 rem 031269 bytes free       ==
1410 rem ========================
```

# TIPS & TRICKS P/4

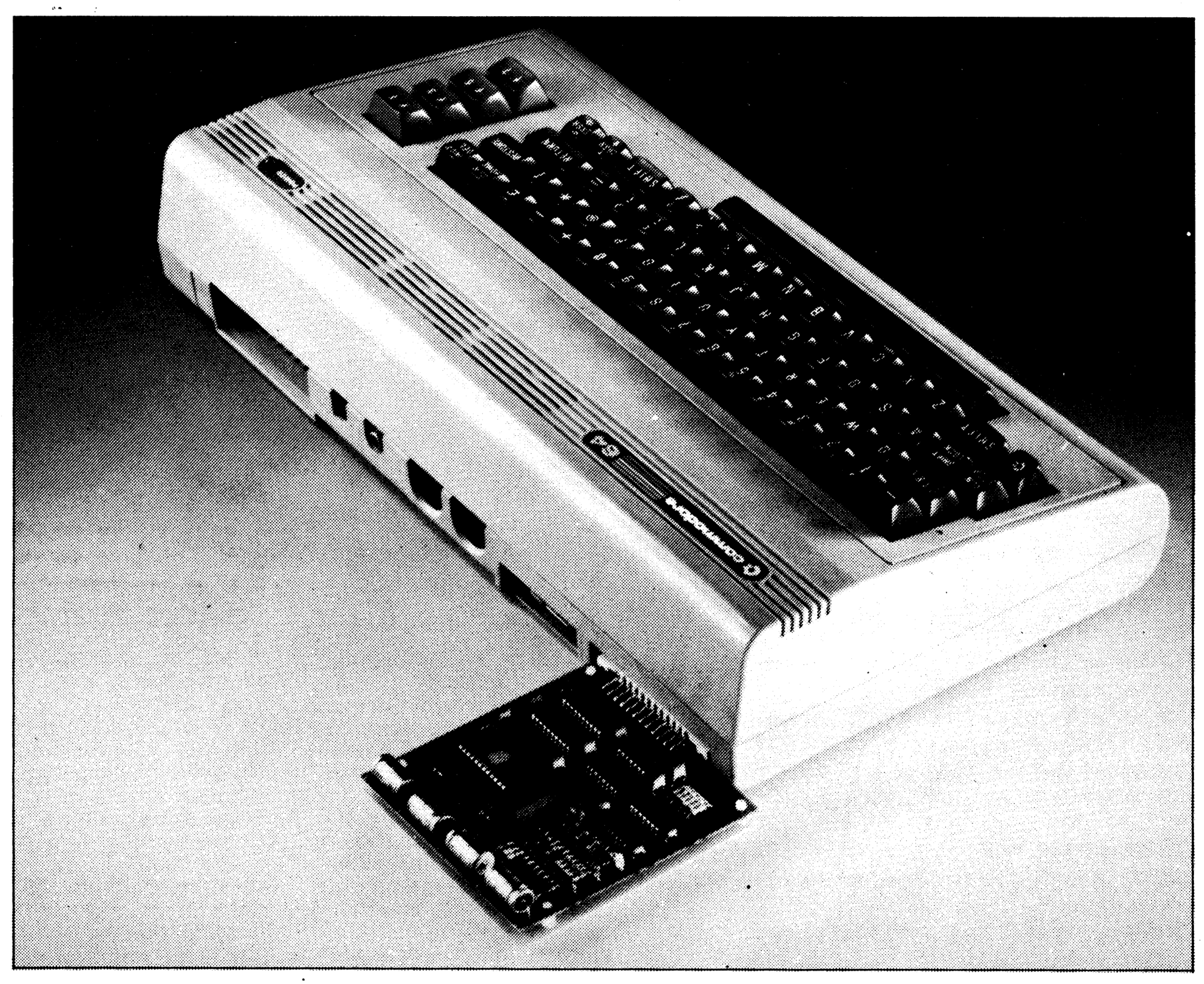
Der 'Plus 4' ist ein bewundernswertes und begeisterndes Gerät und für alle ernsthaften Anwender das Beste, was zu empfehlen ist (Gegensatz: Seitenlange Programme zu Hause auswendig einpauken,um im Warenhaus den Zuschauern etliche bunte Kreise auf den Bildschirm zu machen – gibt es etwas Dümmeres? Kleine Männchen umherhüpfen zu



lassen – ist das im Sinne der Erfinder?).
Gerade die Interessengruppen, für die der Hersteller dieses Gerät gemacht hat, wissen leider nichts von ihm, da sie – zwar interessiert, doch zurückhaltend – in dem noch herrschenden Dschungeldunkel nicht klar sehen und eine Werbung des Herstellers fast völlig fehlt.
Eines der im Plus 4 integrierten und im Anleitungsbuch bestens erklärten Programme ist die sog. Dateiverwaltung. Hierin gibt es den 'RV'-Befehl, um die Dateien schnell durchlaufen zu lassen. 'Schnell' ist hier fast zu schnell, während der zugehörige Stoppbefehl recht langweilig reagiert. Es gibt eine einfache Abhilfe mit dem 'Search'-Befehl, der üblicherweise mit der Eingabe eines Suchbegriffes verbunden ist. In diesem Falle wird das Eingeben eines Suchbegriffes unterlassen und statt dessen ein zweites 'Return" gedrückt. Das bewirkt, daß Nr. 1 der unsortierten oder einer sortierten Datei auf dem Bildschirm erscheint. Gleichzeitig fragt der C.: 'Yes/No?' – bei 'Yes' kommt Nr. 2 der Datei. So kann die ganze Datei-Diskette abgefragt werden – so schnell oder so langsam, wie jeweils 'Yes' gedrückt wird.
Das Anleitungsbuch gibt weiterhin keine Auskunft über den Fall, daß ein einziger Brief, d.h. eine einzige Adresse aus der Dateiverwaltung, verwendet bzw. gedruckt werden soll: Es wird vor der Angabe 'tf;:rc' die RC-Nr. der zu verwendenden Adresse eingetippt, woraufhin der Druckvorgang bei Textende endet, da ja kein 'eof' eingegeben wurde. Andernfalls muß rechtzeitig 'Run/Stop' betätigt werden.
Schließlich ist noch zu erwähnen, daß der Cursor mit 'Shift' und '=' in Spalte 77 gebracht werden kann.
Ein Computer für einen rationalen Zweck ist wesentlich unterhaltsamer als einer nur zur Unterhaltung! *Helmut Kopplin*

# BRENNEN SIE SICH WOHL

**Neben der Software stehen Hardware-Erweiterungen bei VC20/C64-Besitzern immer noch ganz oben auf der Wunschliste. Leider sind auch diese Dinge nicht ganz billig, es sei denn, man ergreift Eigeninitiative und brennt sich eigene Programme in ein EPROM. COMMODORE-WELT stellt einen „EPROM-Brenner" vor.**



**Eprombrenner**

Nachdem man in Heft 2 nachlesen konnte, was Eproms und Eprombrenner sind, so soll hier mal der praktische Einsatz an einem Beispiel gezeigt werden.

Nochmals zur Wiederholung, Eproms sind elektronische Bausteine, welche mit Daten oder Programmen „gefüllt" oder aber auch wieder gelöscht werden können.

Als „Füllgeräte" dient ein sogenannter Eprombrenner, er besteht aus einer elektronischen Schaltung, die Hardware, und aus einem Steuerprogramm, die Software. Angeschlossen werden die Geräte meist am Userport des C64, seltener am Expansionsport. Dorther beziehen sie auch ihre Stromversorgung. Ohne Saft geht da nichts. Löschen kann man diese Eproms, zumindest die aus der 27er Reihe, mittels UV-Licht.

Monitore sind in diesem Zusammenhang keine Bildschirme, sondern Programme. Mit diesen Programmen ist es nun möglich, sich den Speicher des Rechners auf seinen Inahlt hin anzusehen. So ein Monitor verfügt über einen eigenen Befehlssatz. So ist es möglich, Programme an einen anderen Speicherlatz zu schieben, Änderungen vorzunehmen, z.B. an Maschinenprogram-

men, oder gezielt Speicherbereiche auf Disk zu speichern. Ebenso könne Programme an bestimmte Speicherbereiche geladen werden. Dies wird oft angewendet, wenn z.B. Programme mit Autostart verändert werden sollen. Wenn solche und auch andere Programme nicht an die Originaladresse geladen werden, so sind diese nicht lauffähig, und genau das braucht man ja zum „bearbeiten".

## Farben abändern

Und genau so etwas wollen wir heute mal tun. Wir nehmen uns mal das Betriebssystem unseres Rechners (C64) vor. Einen Monitor sollten Sie zu diesem Zweck schon haben und auch einladen. In der Steuersoftware von Eprombrennern ist auch einer „drin", dieser tut es für unsere Zwecke auch. Der Befehlssatz ist im Allgemeinen gleich.

Wenn wir nun am Betriebssystem herumbasteln wollen, so müssen wir wissen, wo dieses sich versteckt hält und was wir eigentlich ändern wollen. Nun, Sie haben vielleicht noch nicht gemerkt, wie sehr Ihnen diese wunderschönen Einschaltfarben auf den Geist gehen. Sie gehen Ihnen auf den Geist, ganz bestimmt. Und genau diese Farben werden wir ganz nach Ihren speziellen Wünschen abändern und zwar dauerhaft. Keine Angst bitte, ich lasse Sie nicht nach den Farbadressen im Speicher bzw. im Betriebssystem suchen. Nachdem das Betriebssystem fest in ein Prom (nicht löschbarer Baustein) einprogrammiert wurde und ständig in Belegung ist, ist es sicherlich einleuchtend, daß wir da an den Originaladressen nichts „drehen" können. Wir müssen also unser Betriebssystem an einen Speicherplatz bringen, wo es schön ruhig liegt und unsere Eingriffe freudig entgegennimmt. Bevor wir uns zu einem solchen Plätzchen entschließen, sollten wir wissen, wo unser Monitor bzw. unser Steuersoftware im Speicher steht. Monitore stehen oft ab C000 hexadezimal und werden mit XYX 49152 (das ist gleiche Adresse, nur dezimal) gestartet. Die Steuersoftware von Brennern benutzen oft den Bereich ab 5000 hexadezimal. Hier, an dieser Stelle, sollten wir vereinbaren, daß, wenn ein $ (Dollarzeichen) vor einer Zahl steht, damit die hexadezimale Schreibweise gemeint ist. Also gehen wir mal davon aus, daß einer der Bereiche zutrifft, so können wir das Betriebssystem, welches 8 kByte lang ist, an die Adresse $ 1000 verschieben. Das Betriebssystem steht ab $ E000 bis FFFF. Bevor wir zu „schieben" anfangen, sollten wir den auserwählten Bereich noch von irgendwelchen Daten säubern. Dazu müssen wir unseren Monitor, egal welcher, starten. Er meldet sich so:

PC SR AC XR YR SP
.:2003 BO AA F4 91 F7

Falls Ihre Meldung anders ausfällt, so machen sie sich keine Gedanken darüber. Anderer Monitor, andere Meldung. Mit dem „F" oder „O" Befehl läßt sich der auserwählte Bereich „säubern". Sie geben ein: F 1000 4000 FF und drücken die „RETURN" Taste, es kann sein, daß bei anderen Monitoren statt Leerzeichen ein Komma zwischen den Zahlen eingegeben werden muß. Nach einiger Zeit meldet sich unser Cursor blinkend wieder, so, als ob nichts geschehen wäre. Es hat sich aber was getan und zwar haben wir den Speicherbereich mit Nullen (FF) aufgefüllt. Wir können das mit dem „M"-Befehl überprüfen, wenn wir M 100 4000 eingeben. Mit „RETURN" abschließen. Jetzt wird sofort der Speicherbereich ausgelistet. Links werden die Speicheradressen, rechts davon in zweier Gruppen die Daten bzw. Speicherinhalt in ASCII-Code, angezeigt. Nachdem wir uns überzeugt haben, daß unser Bereich „sauber" ist, können wir mit dem Verschieben des Betriebssystems beginnen. Dazu dient der „T" (Transfer- oder Verschiebe) Befehl. Wir geben ein:

T E000 FFFF 1000

Mit „RETURN" abgeschlossen tut sich eine Weile lang nichts sichtbares. Dann kommt nach einiger Zeit unser Cursor wieder und blinkt nichtssagend.

## Exakte Kopie im Speicher

Aber nachdem wir unseren „M" Befehl (Memory ansehen) schon kennen, können wir uns ansehen, was da passiert ist. Also geben wir ein:

M 1000 2FFF und „RETURN"

Sofort wird wieder unser Bereich ausgelistet und siehe da, unsere Nullen sind verschwunden. Statt dessen steht jetzt eine exakte Kopie des Betriebssystems im Speicherbereich $ 1000 bis $ 2FFF dieser Bereich entspricht einer Länge von 8 kByte. Wenn Ihnen das Auslisten zu schnell geht, so können Sie mit der „CONTROL" Taste die Ausgabe verlangsamen. Auch können Sie diesen „Hex-Dump" auf dem Drucker laufen lassen, dazu müssen bei manchen Monitoren die Ausgabebefehle „geshiftet" eingegeben werden. Bei anderen müssen vor dem Starten des Monitors der Druckerkanal geöffent und die Ausgaben auf den Drucker geleitet werden. Mit „OPEN1,4:CMD1" ist dies schon gemacht.

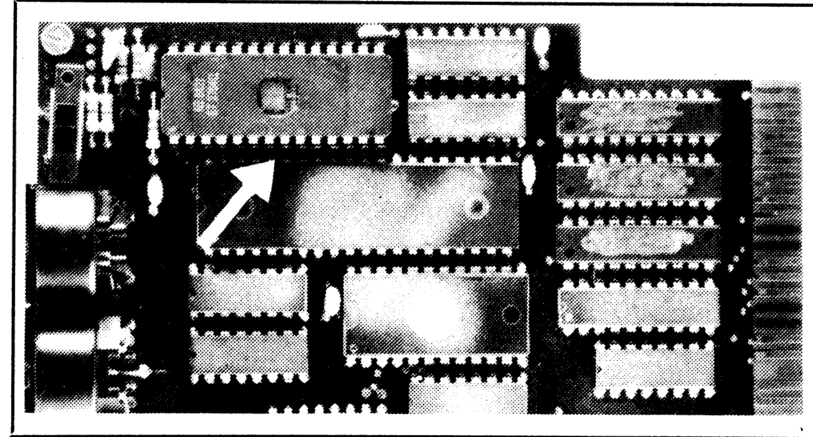
Aber wir wollen ja unsere Einschaltfarben ändern, dazu müssen wir mit dem „M"-Befehl die Speicherzellen $ 1cd9 und $ 1cda auslisten. In der ersten Zelle steht 0E für hellblau, in der zweiten 06 für blau. Also ist der Inhalt von Zelle $ 1cd9 die Farbe vom Rahmen und von Zelle $ 1cd9 die Farbe vom Rahmen und von Zelle $ 1cda die Farbe vom Hintergrund. Wollen Sie z.B. Rahmen und Hintergrund in schwarz haben, so fahren Sie mit dem Cursor auf 0E und geben 90, weiter auf 06, auch hier 90 eingeben, fahren Sie mit dem Cursor auf den Doppelpunkt am Ende dieser Zeile und drücken Sie „RETURN". Wollen Sie die Schriftfarbe gleich mitändern, so listen Sie die Speicherzelle $ 1478 aus und schreiben dort 1E für grün hinein. Dazu ist die gleiche Prozedur wie vorher beschrieben notwendig. auch können Sie die Einschaltmeldung auf die gleiche Art und Weise ändern. Haben Sie alle Änderungen vollzogen, so sollten Sie als erstes Ihre „Kreation" auf Disk speichern. Das geht ebenfalls mit dem Monitor, und mit dem „S"-Befehl (Save) können Sie den Speicherbereich auf Disk bannen. Hierzu müssen Sie Ihrem Befehl aber mitteilen, von wo bis wo abgespeichert werden soll. Ihr geändertes Betriebssystem steht ab $ 1000 bis $ 2FFF im Speicher, also müssen Sie diesen Bereich

plus 1 für die Endadresse abspeichern. Also geben wir ein:
S'KERNAL $ 1000', 08, 1000 3000 und „RETURN"

Somit wäre jetzt alles gesichert und Sie können jederzeit das Betriebssystem mit dem „L"-Befehl (Load) wieder in den Rechner einladen. Dabei wird automatisch an die Adresse geladen, welche beim Abspeichern angegeben wurde, also hier $ 1000. Deshalb auch die Namensgebung mit $ 1000, so weiß man auch nach Jahren noch, wo hingeladen wird.

## Gegen Löschen Schützen!

Ich gehe mal davon aus, daß Sie Ihren Eprombrenner eingesteckt und die Steuersoftware im Rechner haben. So können Sie jetzt mit „X" den Monitor verlassen und ein Menue der Steuersoftware erscheint. Sie sollten jetzt ein leeres Eprom von 8 kByte (2764) in die Fassung des Brenners stecken und den Brenner einschalten. Die weiteren Schritte beziehen sich auf den Eprombrenner von DELA-Elektronik (Bausatz ca. DM 100). Sie wählen den Menuepunkt „Eprom brennen", jetzt fragt sie der Rechner nach dem Epromtyp und Sie geben den Wert, der für 2764 steht, ein. Danach werden Sie aufgefordert, Anfangs- und Endadresse einzugeben. Also hier wieder $ 1000 $ 3000. Die weitere Frage nach dem Bereich im Eprom beantworten Sie einfach mit „RETURN". Jetzt können Sie auf dem Bildschirm den Brennvorgang mitbeobachten. Ist der „Brand" gelungen, so schalten Sie den Brenner aus und entnehmen das Eprom aus seiner Fassung. Wie eingangs erwähnt, lassen sich Eproms mit UV-Licht löschen. Deshalb haben diese Bausteine ein Fenster. Durch dieses kann UV-Licht einfallen. Das Eprom muß also gegen unbeabsichtigtes Löschen geschützt werden. Überkleben Sie das Fenster einfach mit einem Überschreibschutzstreifen von Ihren Disketten. Darauf können Sie auch eine Kennzeichnung, wie z.B. Kernal Farbe, anbringen.


Das fertige Eprom C

Hier noch eine Farbtabelle, damit Sie sich „Ihre" Farbkombination zusammenstellen können. Sie ist zwar auch in Ihrem Handbuch enthalten, allerdings nicht im Hexcode.

| Farbe | ASCII CHR$ | HEX |
|---|---|---|
| weiß | 5 | 05 |
| rot | 28 | 1c |
| grün | 30 | 1e |
| blau | 31 | 1f |
| schwarz | 144 | 90 |
| purpur | 156 | 9c |
| gelb | 158 | 9e |
| cyan | 159 | 9d |

Sie sollten sich in Ihr Handbuch den jeweiligen Hex-code hinter die CHR$ schreiben. Beachten Sie dabei aber die Zählweise im Hexcode und schreiben Sie die Werte auch als 2Byte in Ihre Liste. Also so: CHR$ 0 = Hex 00 und denken Sie daran, nach 9 kommt nicht 10 sondern 0A und nach 0F gehts weiter mit 10 (Eins Null). So, und jetzt müssen Sie Ihr Eprom auch noch in den Rechner einbauen. Dazu benötigen Sie auf jeden Fall den in Heft 2/86 beschriebenen Adapter, denn Ihr Eprom hat ein paar Füßchen mehr als Ihr Prom (ernal) im Rechnr. Im Fachhandel können Sie solche Adapter oder auch Betriebssystemumschaltplatinen bekommen. Bei DEAL in Köln beispielsweise.

Zurück zum Einbau. Öffnen Sie Ihren Rechner und klappen Sie die Tastatur vorsichtig nach hinten, die einzelnen Port sollten auch nach hinten zeigen. Wenn Sie jetzt die Abschirmpappe nach vorne ziehen, können Sie direkt auf die Platine schauen. Links hinten, in der Nähe des Userports, finden sie 2 „längere" IC's, das sind Ein-Ausgabebausteine, rechts daneben finden sie noch 3 kleinere IC's. Alle schön parallel angeordnet. Der Mittlere dieser drei ist der, welcher ausgetauscht werden soll. Wenn Sie Glück haben, ist dieses IC bereits gesockelt. In diesem Falle ziehen Sie das IC vorsichtig aus dem Sockel und stecken Sie nun Ihren Adapter mit Eprom (Kerbe nach vorn, Richtung Userport) in die Fassung. Sollte Ihr Kernal eingelötet sein, so müssen Sie die Computerplatine ausbauen und das Kernal auslöten und einen Sockel einlöten. Achten Sie bitte beim Ausbau der Platine auf verschiedene Kabel, die zum Gehäuseoberteil führen. Auch beim Auslöten ist größte Vorsicht geboten, denn sehr schnell sind die dünnen Leiterbahnen auf der Platine verletzt, auch das IC kann durch zu hohe Temperatur des Lötkolbens beschädigt werden. Am besten ist es, wenn Sie diese Arbeit von einem Fachgeschäft (Radio/Fernseh-) ausführen lassen, denn diese haben Speziallötgeräte und evtl. auch die größere Erfahrung in solchen Dingen.

Bauen Sie Ihren Rechner wieder zusammen, denken Sie an die Kabel, schließen Sie alles wieder an und geben Sie Ihrem Prunkstück „Saft". Jetzt gibt es nur zwei Möglichkeiten: Es geht oder es geht nicht. Im letzteren Falle alles nochmal überprüfen und gegebenenfalls ein neues Eprom brennen. Wenn dies der Fall sein sollte, werden Sie spätestens jetzt auf den Knien liegend Dankeshymnen singen, wenn Sie Ihr Keranal unbeschädigt aus dem Rechner bekommen haben.

Merke: Ohne Kernal kein Leben mehr im Rechner.

# EINFACH: 40 KB PER SOFTWARE

**Ein Aprilscherz hatte Folgen. Zum Beispiel diesen Artikel, der keiner ist.**

Wer die April-Ausgabe der COMMODORE WELT gelesen hat, dem ist sicher unser April-Scherz aufgefallen. Natürlich kann der C64 nicht softwaremäßig mit 1 Mega-Byte Speicher ausgerüstet werden und einen Atari 520 simulieren. Den meisten ist dies wohl bereits beim Lesen des Textes aufgefallen.

Diejenigen, welche gewohnheitsmäßig die Programme abtippen, bevor Sie lesen, wozu die gut sind, werden den Scherz spätestens beim ersten ‚READY' sprich ‚APRIL APRIL' erkannt haben.

Aber so wie die meisten Geschichten einen wahren Hintergrund haben, so hat auch unser Programm einen guten Kern: Sie haben damit tatsächlich mehr Speicher für Basic zur Verfügung. Zuerst aber kurz ein paar Erklärungen zum Programm, damit Sie es leicht für eigene Zwecke einsetzen können.

## Nix mehr mit April

Nachdem das Maschinenprogramm eingelesen und das Betriebssystem und der Interpreter ins RAM kopiert wurden (wo sie dann verändert werden können), wird in Zeile 215 die READY-Meldung in APRIL APRIL umgewandelt und in Zeile 225 der Text, welcher beim Einschalten immer erscheint, verändert. Dieser Text wird dann mit SYS51320 ausgegeben. Wollen Sie den Text verändern, so müssen Sie nur darauf achten, daß er nicht länger wird und in den DATA-Zeilen die entsprechenden ASCII Codes der Zeichen austauschen. SYS 51200 kopiert den Originalzeichensatz vom ROM ins RAM (ab 53248) und verändert diesen dabei, indem jedes Byte mit 60 AND-verknüpft wird.. Das heißt, daß nur die inneren 4 Bit stehen bleiben. Dabei wird das ‚M' noch extra verändert, da es sonst etwas unvollständig aussieht. Wenn der Zeichensatz bei 53248 beginnt, muß der Bildschirmspeicher auch im oberen Viertel des Speichers liegen. SYS51246 verlegt den Bildschirmspeicher nach 49152. So und jetzt wird es interessant:

40137 BASICBYTES FREE!

Solange man mit den oberen Bildschirmen (ab 49152, auch ab 50176 ist möglich, siehe 60 Zeichen-Programm arbeitet, kann man den in Basic nutzbaren Speicher vergrößern. Arbeitet man mit Diskette, kann man den Basicanfang auf 820 verlegen:
POKE820,0:POKE43,53:
POKE44,3
Jetzt hat man 1228 Byte mehr Speicher. Mit Datasette kann der Basicanfang auf 1024 verlegt werden: POKE1024,0:POKE 44,4
Nun, das ist zwar kein Mega-Byte, sondern nur etwas mehr als ein Kilo-Byte, aber das ist doch auch schon etwas. Damit aber noch nicht genug. Sie haben nun auch noch ein originelles Werkzeug, um in Windeseile den Zeichensatz zu verändern. Sie haben ja gesehen, wie der Zeichensatz aussieht, wenn man durch ‚AND 60' nur 4 Byte übernimmt. Fügt man folgende Zeile ein, so hat man den Originalzeichensatz:
122 POKE 51218,255:
POKE 51345,96
Nachdem das Programm nun gestartet wurde, geben Sie mal im Direktmodus ein:
FOR I=1 TO 255: POKE
51218,I:SYS51200:
PRINT
CHR$(19)I:WAIT198,1:
POKE198,0:NEXT
Sie können jetzt auf Tastendruck alle 255 Möglichkeiten durchspielen und links oben die entsprechende Zahl in 51218 ablesen. Suchen Sie sich eine aus und freuen sich am zusätzlichen Speicher. Zum normalen Zustand kommt man mit:
POKE56576,152:POKE
53272,21:POKE648,4:
POKE1,55

## Blockverschiebung

```
1 rem blockverschieberoutine
2 fori=828to860:reada:pokei,a:nexti
4 data173,93,3,174,94,3,133,95,134,
96
6 data173,95,3,174,96,3,133,90,134,
91
8 data173,97,3,174,98,3,133,88,134,
89,76,191,163
10 rem interpreter ins ram
12 fori=861to866:pokei,0:nexti
14 poke862,160:poke864,192:poke866,
192:sys828
15 rem kernal ins ram
16 fori=861to866:pokei,0:nexti
18 poke862,224:sys828
90 rem korrektur rom 2 -> rom 3
100 fori=58579to58588:readx:pokei,x
:next:fori=58748to58777:readx:pokei
,x:next:fori=59911to59922:readx:pok
ei,x:next:poke58540,129:poke58914,1
45:poke58915,229:poke61332,76:poke6
1333,211:poke61334,228:poke65408,3:
data133,169,169,1,133,171,96,173,13
4,2
111 data32,240,233,169,39,232,180,2
17,48,6,24,105,40,232,16,246,133,21
3:data76,36,234,228,201,240,3,76,23
7,230,96,234:data32,218,228,169,32,
145,209,136,16,246,96,234
200 poke1,5:print"Skernal und inter
preter sind jetzt im ram"
```

## Freude am zusätzlichen Speicher

# LISTSCHUTZ FÜR PROFIS

**Unsere Serie über die Listschutzmethoden hat bei den Lesern großen Anklang gefunden. Hier ist der letzte Teil. Nehmen Sie sich also noch einmal das Heft 4/86 zur Hand und frischen Sie Ihre Kenntnisse über Listschutzmethoden auf.**

Wird einem Farb- und Steuercode zwischen 128 und 160 ein 141 vorgestellt, so wird zunächst das SHIFT/RETURN ausgeführt und die darauffolgenden Steuerzeichen stehen alleine auf der nächsten Zeile. Und prompt werden sie auch ausgeführt, was ohne 141 unmöglich ist. Es geht auch mit dem Code 13 (RETURN).

Das wollen wir doch mal näher betrachten: Geben Sie bitte ein:

```
1PRINT":(CLEAR) (6*DOWN) (6*RIGHT) (BLACK)HALLO(BLUE) (6*UP)"
```

Anstelle der Klammerausdrücke muß die entsprechende Taste gedrückt werden. Zum Beispiel für (CLEAR) drückt man SHIFT/CLR HOME. Nun noch folgendes eingeben:

```
POKE 2055,141:LIST
```

Es steht ein schwarzes HALLO einsam am Bildschirm und sonst scheint sich nichts mehr zu tun. Die READY-Meldung und jede Eingabe ist jetzt wegen der blauen Farbe unsichtbar. Mit diesem Trick kann man also Listings verschwinden lassen oder beliebig einfärben (Zum Verschönern von Listings mit solchen und ähnlichen Methoden habe ich ein Programm (Listingchanger) geschrieben, das in der CBM-REVUE 5/85 S. 42 erschienen ist.) Der aufmerksame Leser hat sicher bemerkt, daß uns noch die Codes kleiner 32 fehlen. In Tabelle 1 sind alle Steuercodes zusammengefaßt. Besondere Beachtung verdienen noch die 13, die 19, die 20 und die 31. Das Tolle an den Codes unter 32 ist die Tatsache, daß sie auch beim Listen ausgeführt werden, 13 und 20 sogar innerhalb von Anführungszeichen. Vielleicht kennen Sie die Methode, wie man mit dem Code 20 (DEL) Zeilen oder Teile davon unsichtbar machen kann.

Hier ein Beispiel:

```
1 PRINT „TEST":GOTO 1:REM"TTTTTTTTTTT
```

Hierzu muß man (nur für die, die es noch nicht wissen, nach REM 2 mal Anführungszeichen, 1 mal DEL, 12 mal SHIFT/INST und 12 mal DEL



Diskettenknacken? Jetzt nicht mehr

drücken (diese erscheinen als in-verse T). Durch diese Manipulation ist das GOTO 1 verschwunden. Interessant ist die Kombination mit REMSHIFT/L:
**1 REM**
**2 REM"TTTTTTT"L**
**(L=SHIFT/L)**

Dadurch, daß ein DEL-Code mehr eingegeben wurde, als zum Löschen von REM und Zeilennummer nötig, wandert das zweite Anführungszeichen (wichtig, da sonst SHIFT /L als Grafikzeichen gelistet wird und keinen SYNTAX ERROR hervorruft) an das Ende der Zeile 1, so daß es aussieht, als wäre der Fehler in Zeile 1. Anstelle der Gänsefüßchenmethode kann man die 20 auch in vorher reservierte Speicherstellen POKEn.

Eine andere Möglichkeit, den „Gegner" zum Löschen einer falschen Zeile zu verlocken, ist der RETURN-Coe (13):
**10 REM TEST*20 REM L (wieder SHIFT/L!)**
das Sternchen überPOKEn wir jetzt mit POKE 2062,13. Dadurch erscheint beim Listen auf dem Bildschirm:
**10 REM**
**20 REM**
**SYNTAXERROR**

Der Versuch, Zeile 20 zu löschen, ist natürlich sinnlos. Dabei wird nur die tatsächliche Zeile 20 gelöscht.

Durch
**0 REM (RETURN)**
**POKE 2054,31:POKE 2051,255:POKE2052, 255**
kann man das ganze Listing blau einfärben und somit unsichtbar machen. Hierbei darf das Zurücksetzen der Zeilennummer im Programm nicht vergessen werden.

Um zu verhindern, daß ein geschütztes Programm nach irgendwelchen Änderungen, wie z.B. dem Entfernen des Copyrights, noch lauffähig bleibt, gibt es wiederum verschiedene Methoden.

Da ist zunächst eine im Programm versteckte Abfrage nach dem Inhalt bestimmter Speicherstellen, wie z.B. 45 und 46, oder FRE(0).
Zum Beispiel:
**1000 if fre(0) ⟨ ⟩ xxxxs then new**

Für xxxxx wird der Wert eingesetzt, den man bei fertigem Programm mit ?fre(0) erhält.
**IF GOTO=END THEN AND=OR????**

Eine sehr interessante und weniger bekannte Methode ist die Verwendung von „unmöglichen" Variablennamen wie ST, END, TO, IF ... Versuchen Sie einmal, der Variablen ST den Wert 3 zuzuordnen:
**1 ST = 3.?ST**

Das geht normalerweise deshalb nicht, weil ST, IF ... zu den reservierten Wörtern gehören. Auch an dieser Stelle kann man den Computer „überlisten". Tippt man innerhalb eines reservierten Wortes einen geshifteten Buchstaben ein, wo wird das Wort nicht erkannt und als Variablename eingestuft:
**1 E(SHIFT/E)ND = +: PRINT E(SHIFT/E)ND**
ergibt beim Listen:
**1 END = 3:PRINT END**
und mit RUN wird END als normale Variable behandelt. Wenn Sie aber mit dem Cursor auf die Zeile fahren und RETURN drücken, läuft es natürlich nicht mehr (Sehen Sie sich mal mit dem MINIMONITOR den Speicherinhalt in beiden Fällen an.). Es ist nicht schwer, sich nun mit diesen Kenntnissen einige Fallstricke durch Kombination der Möglichkeiten selbst auszudenken.

Doch Halt. Wir haben noch etwas sehr wichtiges vergessen. Was machen wir, wenn der „unbefugte Lister" einen Drucker besitzt?

Hier muß man sich ein wenig mit der Druckeransteuerung beschäftigen. Der Code 8 erzeugt am Bildschirm eine Verriegelung der Umschaltung zwischen Groß- und Kleinschrift. Beim Drukker wird dadurch in den Grafik-Modus umgeschaltet, und er erwartet eine Grafikinformation. Folgt auf die 8 eine Zahl kleiner 128, so bricht der Drucker das Listing ab. Außer durch hineinPOKEn in den entsprechenden Speicherbereich kann man den Coe 8, welcher auf dem Schirm als reverses H erscheint (nach "), durch folgende Methode einsetzen:
**10 rem test" (CTRL/H)"**

Sie merken die Wirkung dadurch, daß sich nach dem Listen der Groß/Kleinmodus nicht mehr einstellen läßt.

## Das stoppt jeden Drucker

Leider gibt es aber Drucker, die sich um nicht druckbare Zeichen gar nicht kümmern.

Daher zeigen wir jetzt noch eine wesentlich bessere Methode, welche unseres Wissens immer funktioniert:
**10 :::::PRINT ABC**
**PIKE 2053,0:LIST**

Dadurch steht am Anfang der Basiczeile eine Null, wodurch der Computer ‚denkt', daß sie hier auch schon zuende ist. Beim Programmdurchlauf werden jedoch die vier auf die Null folgenden Zeichen übersprungen. Hier haben wir nun einen Listschutz, der (endlich) sowohl auf dem Schirm als auch auf dem Drucker funktioniert.

Un damit kommen wir zum Ende der uns bekannten Listschutzmethoden. Man kann auch noch mit Veränderungen der Basicendeadressen in 45 und 46 experimentieren. Zum Beispiel, indem man die drei Nullen am Basicende wegPOKEt und die Adresse in 45 und 46 erhöht. Im Programm müssen sie dann wieder zusammen mit der letzten Linkadresse vor der letzten Zeile wiederhergestellt werden. Die Ergebnisse dieser Methode sind aber stark vom Inhalt der Speicherzellen vor dem Laden des Programms abhängig und schinen uns nicht so empfehlenswert.

## Selbermachen ist der Sinn

Last but not least:

Man kann sein Programm auch noch verschlüsseln. Eine wirksame Methode dazu wurde in der CBM-REVUE 2/86 beschrieben. Hierbei wird der gesamte Speicherinhalt durch eine EOR-Verknüpfung verändert, so daß das Programm nur noch mit einem Codewort zum Laufen zu bringen ist. Zum reinen Listschutz eignet sich die Methode weniger, denn will man das Programm (nur zur Benutzung) weitergeben, so muß man auch das Paßwort preisgeben und damit kann man das Programm ja entschlüsseln. Jede Listschutzmethode ist nur solange wirksam, wie sie niemand kennt. Aber auch mit diesen, nun allen CBM-Lesern bekannten, Tricks kann man durch Kombination verschiedener Methoden einen sehr wirksamen Schutz aufbauen, der zum Knacken fast soviel Zeit und Kenntnisse erfordert, wie das Programm selbst zu schreiben. Und das ist ja der Sinn des Ganzen.
(rsf)

# MATHE-TRAINER

Nach dem Starten des Programmes wird man zur Eingabe seines Namens und des aktuellen Datums aufgefordert. Diese Angaben werden beim Ausdruck der Gesamtauswertung gebraucht.
Dann erscheint das Menü.
Addition
Subtraktion
Multiplikation
Division
Gesamtauswertung
Level ändern
Programm beenden
Rechts neben dem ersten Menüpunkt ist ein Pfeil (←) zu sehen, der mit den Funktionstasten F1 und F7 auf und ab bewegt werden kann.
Steht er neben dem gewünschten Punkt, wählt man diesen mittels RETURN an.

### 1.) Addition

Nach Anwahl dieses Menüpunktes wird man aufgefordert, eine Taste zu drücken. Ist dies erfolgt, erscheint die zu lösende Aufgabe und die Zeitzählung beginnt.
Nach Eingabe des Ergebnisses zeigt der Computer, ob dieses falsch oder richtig war und gibt entsprechend die richtige Lösung aus.
Dann wartet er.
Nun kann man folgende Tasten drücken:
1. SPACE: die nächste Aufgabe wird gestellt
2. ← : der Computer kehrt ins Menü zurück
3. g : die Gesamtwertung wird ausgegeben.

Der Computer reagiert entsprechend auf einen Tastendruck.

### 2.) Subtraktion

Bedienung wie bei "Addition", nur werden diesmal Subtraktionsaufgaben gestellt.

### 3.) Multiplikation

Bedienung wie bei "Addition", nur werden diesmal Multiplikationsaufgaben gestellt.

### 4.) Division

Die Division ist so zu bedienen wie "Addition", allerdings sind je nach Schwierigkeitsgrad ein paar Dinge zu beachten:
Wenn man mit Nachkommastellen rechnen will, muß das Ergebnis entweder auf 2 Nachkommastellen gerundet eingegeben werden oder es müssen 3 oder mehr Nachkommastellen eingegeben werden.
Rechnet man in höheren Schwierigkeitsgraden ohne Nachkommastellen, kann es zu längeren Wartezeiten kommen, da der Computer zwei bis zu vierstellige Zahlen finden muß, die ohne Nachkommastellen dividierbar sind.

ch$ Bildschirm löschen (=chr$(147))
d Feldnummer für Durchschnittszeit
da$ aktuelles Datum
fa Fehleranzahl bei Additionsaufgaben
fd Fehleranzahl bei Divisionsaufgaben
fm Fehleranzahl bei Multiplikationsaufgaben
fs Fehleranzahl bei Subtraktionsaufgaben
h$ =chr$(186)
i Schleifenzähler
j Schleifenzähler
l$ eingegebenes Ergebnis des Benutzers
l1 eingegebenes Ergebnis als Zahl
l2 Divisionsergebnis auf zwei Nachkommastellen gerundet
le derzeitiger Schwierigkeitsgrad (1–4)
na$ Name des Anwenders
nd wird zur Leveländerung benötigt
nl wird zur Leveländerung benötigt
o wird bei Ausgabe der Zeitentabelle gebraucht
re$ Cursor rot (=chr$ 28))
s Feldnummer für Schlechtestzeit
s$ Einheit bei der Zeitausgabe (Sekunde/ Sekunden)
sP Spaltennummer für Cursorposition
st Statusvariable
t$ wird für die Hardcopyroutine benötigt
t Zeit,die bis zur Eingabe verbraucht wurde
t1 auf 2 Stellen gerundete Eingabezeit
t2$ wird für die Hardcopyroutine benötigt
t 2 Durchschnittszeit
ti Variable zur Zeitmessung
w Wertebereich für z1 und z2
wh$ Cursor weiß (=chr$(5))
z richtiges Ergebnis einer Aufgabe
z1 1. Zufallszahl
z2 2. Zufallszahl
z3 Divisionsergebnis
ze Zeilennummer für Cursorposition
zt Speicher für Best-, Schlechtest- und Durchschnittszeit (wird als Feldvariable auf 48 dimensioniert

Wem die Wartezeit zu lang ist, der kann sich durch Druck der Taste ← ins Menü zurückversetzen lassen.

### 5.) Gesamtauswertung

Sobald dieser Menüpunkt angewählt wurden, wird der Benutzer gefragt, ob die Ausgabe auf Drucker erfolgen soll.
Tippt man ein "j" und hat den Drucker nicht eingeschaltet, merkt das der Computer und fordert auf, diesen Fehler durch Einschalten des Druckers zu beheben oder eine Taste zu drücken, wobei dann auf eine Druckerausgabe verzichtet wird.

### 6.) Level ändern

Der auf dem Bildschirm angezeigte Level kann mit den Funktionstasten F1 und F7 verändert werden.
Mit F3 kann man, falls erwünscht, ein negatives Ergebnis bei Subtraktionsaufgaben zulassen.
Mit F5 kann man, wenn erwünscht, Nachkommastellen bei Divisionsaufgaben zulassen.
Mit RETURN kommt man ins Menü zurück.

### 7.) Programm beenden

Nach Anwahl dieses Menüpunktes löscht sich das Programm von selbst (mittels SYS 64738).

### VARIABLENLISTE:

a$ wird bei 'get'-Abfragen verwendet
a hat den Wert .1E-27 und wird bei der Gesamtauswertung gebraucht
aa Anzahl bisher gefragter Additionsaufgaben
ad Anzahl bisher gefragter Divisionsaufgaben
am Anzahl bisher gefragter Multiplikationsaufgaben
as Anzahl bisher gefragter Subtraktionsaufgaben
au Aufgabenart (1=Addition; 2=Subtraktion; 3=Multiplikation; 4=Division)
b Feldnummer für Bestzeit

```
10 rem mathetrainer==============64
20 rem (p) commodore welt        ==
30 rem ===========================
40 rem   (c) by                  ==
50 rem   erik jacobi             ==
60 rem                           ==
70 rem version 2.0  40z/ascii    ==
80 rem c-64 + 1541               ==
90 rem ===========================

100 :
110 clr
120 poke53280,6:poke53281,6:printch
r$(14)chr$(8)
130 ch$=chr$(147):wh$=chr$(5):re$=c
hr$(28):h$=chr$(186)
140 le=1:nl=1:nd=1:a=.1e-27:dimzt(4
8),t2(16),aa(16),as(16),am(16),ad(1
6)
150 printwh$chr$(147):poke19,64:inp
ut"Ihr Name :";na$
160 ze=2:sp=0:gosub3090:input"Heuti
ges Datum :";da$
170 :
180 rem ==========================
190 rem titelbild/menueauswahl   =
200 rem ==========================
210 ifle=1thenw=9
220 ifle=2thenw=99
230 ifle=3thenw=999
240 ifle=4thenw=9999
250 sp=30:poke646,3:poke53280,6:pok
e53281,6
260 printch$tab(10)"xxxxx
270 printtab(10)"x x xathe -
280 printtab(10)"x x x
290 print
300 printtab(10)"trainer
310 print
320 printtab(8)"***"
330 printtab(8)"*C*1985 by..Erik..J
acobi
340 printtab(8)"***"
350 printre$spc(7)"***************
***********
360 printspc(7)"******************
*******
370 poke646,3:print:print:printtab(
10)"Addition...........
380 printtab(10)"Subtraktion.......
.
390 printtab(10)"Multiplikation....
.
400 printtab(10)"Division..........
.
410 printtab(10)"Gesamtauswertung..
.
420 printtab(10)"Level
aendern......
430 printtab(10)"Programm
beenden...
440 ze=13:gosub3090:print"←"
450 geta$:ifa$=""then450
460 ifa$=chr$(13)then520
470 ifa$=chr$(136)thengosub3090:pri
nt" ":ze=ze+1:gosub3090:print"←"
480 ifa$=chr$(133)thengosub3090:pri
nt" ":ze=ze-1:gosub3090:print"←"
490 ifze=20thengosub3090:print" ":z
e=13:gosub3090:print"←"
500 ifze=12thengosub3090:print" ":z
e=19:gosub3090:print"←"
510 goto450
520 ifze=13then680
530 ifze=14then980
540 ifze=15then1280
550 ifze=16then1580
560 ifze=17then1910
570 ifze=18then2750
580 ifze=19then3040
590 :
600 :
610 :
620 rem ========================
630 rem = addition             =
640 rem ========================
650 :
660 :
670 :
680 poke53280,2:poke53281,2:s$="Sek
unde":aa=aa+1:au=1:gosub3130:aa(t3)
=aa(t3)+1
690 printch$chr$(150)"************
**************************"
700 printchA$(145)"*"spc(14)chr$(15
8)"Addition"spc(16)chr$(150)"*"
710 printchr$(145)"***************
***********************"
720 z1=int(rnd(1)*w)+1
730 z2=int(rnd(1)*w)+1:z=z1+z2
740 print:print:printtab(15)"******
*"
750 printtab(15)"*"chr$(158)"TASTE"
chr$(150)"*"
760 printtab(15)"*******"
770 poke198,0:wait198,1:poke198,0
780 t=ti:print:print:print"Aufgabe:
 "z1" + "z2
790 poke19,64:printchr$(5):input"Lo
esung: ";l$:poke19,0
800 sp=0:ze=15:gosub3090
810 ifval(l$)=zthenprintchr$(150)"R
ICHTIG"
820 ifval(l$)<>zthenprintchr$(150)"
FALSCH":fa=fa+1
830 ifval(l$)<>zthenprint:printchr$
```

```
(5)"Richtige Loesung: ";z
84Ø t=(ti-t)/66:t1=int(t*1ØØ+.5)/1Ø
Ø
85Ø ift1>1ort<1thens$=s$+"n"
86Ø print:printchr$(144)"Benoetigte
 Zeit: "t1" "s$:t2(t3)=t2(t3)+t1
87Ø zt(d)=int((t2(t3)/aa(t3))*1ØØ+.
5)/1ØØ
88Ø gosub65535
89Ø geta$:ifa$=""then89Ø
9ØØ ifa$="←"then21Ø
91Ø ifa$=" "then68Ø
92Ø ifa$="g"then191Ø
93Ø goto89Ø
94Ø rem ========================
95Ø rem subtraktion            =
96Ø rem ========================
97Ø :
98Ø poke5328Ø,2:poke53281,2:s$="Sek
unde":as=as+1:au=2:gosub313Ø:as(t3)
=as(t3)+1
99Ø printch$chr$(15Ø)"************
***************************"
1ØØØ printchr$(145)"*"spc(12)chr$(1
58)"Subtraktion"spc(15)chr$(15Ø)"*"
1Ø1Ø printchr$(145)"**************
************************"
1Ø2Ø z1=int(rnd(1)*w)+1
1Ø3Ø z2=int(rnd(1)*w)+1:z=z1-z2
1Ø4Ø ifn1=1andsgn(z)=-1then1Ø2Ø
1Ø5Ø print:print:printtab(15)"*****
**"
1Ø6Ø printtab(15)"*"chr$(158)"TASTE
"chr$(15Ø)"*"
1Ø7Ø printtab(15)"*******"
1Ø8Ø poke198,Ø:wait198,1:poke198,Ø
1Ø9Ø t=ti:print:print:print"Aufgabe
: "z1" - "z2
11ØØ poke19,64:printchr$(5):input"L
oesung: ";l$:poke19,Ø
111Ø sp=Ø:ze=15:gosub3Ø9Ø
112Ø ifval(l$)=zthenprintchr$(15Ø)"
RICHTIG"
113Ø ifval(l$)<>zthenprintchr$(15Ø)
"FALSCH":fs=fs+1
114Ø ifval(l$)<>zthenprint:printchr
$(5)"Richtige Loesung: ";z
115Ø t=(ti-t)/66:t1=int(t*1ØØ+.5)/1
ØØ
116Ø ift1>1ort<1thens$=s$+"n"
117Ø print:printchr$(144)"Benoetigt
e Zeit: "t1" "s$:t2(t3)=t2(t3)+t1
118Ø zt(d)=int((t2(t3)/as(t3))*1ØØ+
.5)/1ØØ
119Ø gosub65535
12ØØ geta$:ifa$=""then12ØØ
121Ø ifa$="←"then21Ø
122Ø ifa$=" "then98Ø
123Ø ifa$="g"then191Ø
124Ø goto12ØØ
125Ø rem ======================
126Ø rem multiplikation       =
127Ø rem ======================
128Ø poke5328Ø,2:poke53281,2:s$="Se
kunde":am=am+1:au=3:gosub313Ø:am(t3
)=am(t3)+1
129Ø printch$chr$(15Ø)"***********
****************************"
13ØØ printchr$(145)"*"spc(11)chr$(1
58)"Multiplikation"spc(13)chr$(15Ø)
"*"
131Ø printchr$(145)"**************
************************"
132Ø z1=int(rnd(1)*w)+1
133Ø z2=int(rnd(1)*w)+1:z=z1*z2
134Ø print:print:printtab(15)"*****
**"
135Ø printtab(15)"*"chr$(158)"TASTE
"chr$(15Ø)"*"
136Ø printtab(15)"*******"
137Ø poke198,Ø:wait198,1:poke198,Ø
138Ø t=ti:print:print:print"Aufgabe
: "z1" * "z2
139Ø poke19,64:printchr$(5):input"L
oesung: ";l$:poke19,Ø
14ØØ sp=Ø:ze=15:gosub3Ø9Ø
141Ø ifval(l$)=zthenprintchr$(15Ø)"
RICHTIG"
142Ø ifval(l$)<>zthenprintchr$(15Ø)
"FALSCH":fm=fm+1
143Ø ifval(l$)<>zthenprint:printchr
$(5)"Richtige Loesung: ";z
144Ø t=(ti-t)/66:t1=int(t*1ØØ+.5)/1
ØØ
145Ø ift1>1ort<1thens$=s$+"n"
146Ø print:printchr$(144)"Benoetigt
e Zeit: "t1" "s$:t2(t3)=t2(t3)+t1
147Ø zt(d)=int((t2(t3)/am(t3))*1ØØ+
.5)/1ØØ
148Ø gosub65535
149Ø geta$:ifa$=""then149Ø
15ØØ ifa$="←"then21Ø
151Ø ifa$=" "then128Ø
152Ø ifa$="g"then191Ø
153Ø goto149Ø
154Ø rem ======================
155Ø rem = division           =
156Ø rem ======================
157Ø :
158Ø poke5328Ø,2:poke53281,2:s$="Se
kunde":ad=ad+1:au=4:gosub313Ø:ad(t3
)=ad(t3)+1
159Ø printch$chr$(15Ø)"***********
****************************"
16ØØ printchr$(145)"*"spc(14)chr$(1
58)"Division"spc(16)chr$(15Ø)"*"
```

```
1610 printchr$(145)"**************
************************"
1620 z1=int(rnd(1)*w)+1
1630 z2=int(rnd(1)*w)+1:z=z1/z2:z3=
z:z=int(z*100+.5)/100
1640 geta$:ifa$="←"thenad=ad-1:ad(t
3)=ad(t3)-1:goto210
1650 ifnd=1andz3<>int(z3)then1620
1660 print:print:printtab(15)"*****
**"
1670 printtab(15)"*"chr$(158)"TASTE
"chr$(150)"*"
1680 printtab(15)"*******"
1690 poke198,0:wait198,1:poke198,0
1700 t=ti:print:print:print"Aufgabe
: "z1" / "z2
1710 poke19,64:printchr$(5):input"L
oesung: ";l$
1720 sp=0:ze=15:gosub3090
1730 l1=val(l$)
1740 l2=int(l1*100+.5)/100
1750 ifl2=zthenprintchr$(150)"RICHT
IG"
1760 ifl2<>zthenprintchr$(150)"FALS
CH":fd=fd+1
1770 ifl2<>zthenprint:printchr$(5)"
Richtige Loesung: ";z
1780 t=(ti-t)/66:t1=int(t*100+.5)/1
00
1790 ift1>1ort<1thens$=s$+"n"
1800 print:printchr$(144)"Benoetigt
e Zeit: "t1" "s$:t2(t3)=t2(t3)+t1
1810 zt(d)=int((t2(t3)/ad(t3))*100+
.5)/100
1820 gosub65535
1830 geta$:ifa$=""then1830
1840 ifa$="←"then210
1850 ifa$=" "then1580
1860 ifa$="g"then1910
1870 goto1830
1880 rem ========================
1890 rem = gesamtauswertung      =
1900 rem ========================
1910 poke53280,2:poke53281,2
1920 printch$wh$"Ausgabe auf Drucke
r (j/n) ?"
1930 geta$:ifa$=""then1930
1940 ifa$="j"thend=1:goto1970
1950 ifa$="n"thend=0:goto2030
1960 goto1930
1970 open4,4:poke768,185:print#4:cl
ose4:poke768,139
1980 ifst<>-128then2030
1990 ze=5:sp=0:gosub3090
2000 print"Bitte den Drucker einsch
alten oder Taste druecken !!"
2010 geta$:ifa$<>""then2030
2020 goto1970
2030 poke53280,2:poke53281,2
2040 printch$chr$(150)"************
******************************"
2050 printchr$(145)"*"spc(10)chr$(1
58)"Gesamtauswertung"spc(12)chr$(15
0)"*"
2060 printchr$(145)"**************
************************"
2070 printchr$(158)
2080 print"Gestellte Additionsaufga
ben: "aa
2090 print"Davon falsch: "fa
2100 print"Fehlerquote: "int((fa/(a
a+a))*100)"%"
2110 print:print"Gestellte Subtrakt
ionsaufgaben: "as
2120 print"Davon falsch: "fs
2130 print"Fehlerquote: "int((fs/(a
s+a))*100)"%"
2140 print:print"Gestellte Multipli
kationsaufgaben: "am
2150 print"Davon falsch: "fm
2160 print"Fehlerquote: "int((fm/(a
m+a))*100)"%"
2170 print:print"Gestellte Division
saufgaben: "ad
2180 print"Davon falsch: "fd
2190 print"Fehlerquote: "int((fd/(a
d+a))*100)"%"
2200 poke198,0:wait198,1:poke198,0
2210 ifd=1andst<>-128thengosub2600
2220 printch$tab(5)chr$(158)"ZEITUE
BERSICHT (in Sekunden)"
2230 print"************************
*************"
2240 print"*L* Add...* Sub...* Mul.
..* Div...*..*"
2250 print"************************
*************"
2260 ze=4:sp=0:l$="* *":gosub3100
2270 sp=34:gosub3090:print"*Bz*"
2280 ze=5:sp=0:l$="*1*":gosub3100
2290 sp=34:gosub3090:print"*Dz*"
2300 ze=6:sp=0:l$="* *":gosub3100
2310 sp=34:gosub3090:print"*Sz*"
2320 print"************************
*************"
2330 ze=8:sp=0:l$="* *":gosub3100
2340 sp=34:gosub3090:print"*Bz*"
2350 ze=9:sp=0:l$="*2*":gosub3100
2360 sp=34:gosub3090:print"*Dz*"
2370 ze=10:sp=0:l$="* *":gosub3100
2380 sp=34:gosub3090:print"*Sz*"
2390 print"************************
*************"
2400 ze=12:sp=0:l$="* *":gosub3100
2410 sp=34:gosub3090:print"*Bz*"
2420 ze=13:sp=0:l$="*3*":gosub3100
```

```
2430 sp=34:gosub3090:print"*Dz*"
2440 ze=14:sp=0:l$="* *":gosub3100
2450 sp=34:gosub3090:print"*Sz*"
2460 print"*************************
*************"
2470 ze=16:sp=0:l$="* *":gosub3100
2480 sp=34:gosub3090:print"*Bz*"
2490 ze=17:sp=0:l$="*4*":gosub3100
2500 sp=34:gosub3090:print"*Dz*"
2510 ze=18:sp=0:l$="* *":gosub3100
2520 sp=34:gosub3090:print"*Sz*"
2530 print"*************************
*************":o=0
2540 poke198,0:wait198,1:poke198,0
2550 ifd=1andst<>-128thengosub2680
2560 d=0:goto210
2570 rem ===========================
2580 rem = hardcopyroutine         =
2590 rem ===========================
2600 printchr$(144)
2610 open1,4,7:print#1,chr$(14)spc(
4)"MATHETRAINER"
2620 print#1,chr$(15)spc(8)"(c) 198
5 by Erik Jacobi"
2630 print#1,chr$(15)spc(8)"-------
-----------------"
2640 print#1,chr$(14)spc(2)"GESAMTA
USWERTUNG"
2650 print#1,chr$(15)spc(4)"-------
-------------------------"
2660 print#1,chr$(15)"Fuer :";na$
2670 print#1,"Datum :";da$:close1
2680 printchr$(144)chr$(19);:open1,
4,7:open2,3
2690 fori=1to22:forj=1to20:get#2,t$
2700 printchr$(20);:get#2,t2$:print
chr$(157)chr$(148)t$t2$;
2710 print#1,t$t2$;:next:print#1:ne
xt:close1:close2:return
2720 rem ========================
2730 rem = level aendern        =
2740 rem ========================
2750 poke53280,10:poke53281,10
2760 printch$chr$(28)"*************
***************************"
2770 printchr$(145)"*"spc(12)chr$(1
58)"Level aendern"spc(13)chr$(28)"*
"
2780 printchr$(145)"***************
*************************"
2790 sp=16:ze=11:gosub3090:print"**
***"
2800 sp=10:ze=12:gosub3090
2810 printchr$(158)"Level:";chr$(28
);"*";chr$(158);le;chr$(28);"*"
2820 sp=16:ze=13:gosub3090:print"**
***"
2830 sp=0:ze=15:gosub3090:print"Neg
ative Ergebnisse bei Subtraktion";c
hr$(158)
2840 ifnl=0thensp=0:ze=16:gosub3090
:print"ja.."
2850 ifnl=1thensp=0:ze=16:gosub3090
:print"nein"
2860 printchr$(28)
2870 sp=0:ze=18:gosub3090:print"Nac
hkommastellen bei Division";chr$(15
8)
2880 ifnd=0thensp=0:ze=19:gosub3090
:print"ja.."
2890 ifnd=1thensp=0:ze=19:gosub3090
:print"nein"
2900 geta$:ifa$=""then2900
2910 ifa$=chr$(13) then210
2920 ifa$=chr$(133)andle=4thenle=1:
goto2800
2930 ifa$=chr$(133)andle<4thenle=le
+1:goto2800
2940 ifa$=chr$(136)andle=1thenle=4:
goto2800
2950 ifa$=chr$(136)andle>1thenle=le
-1:goto2800
2960 ifa$=chr$(134)andnl=1thensp=0:
ze=16:gosub3090:print"ja  ":nl=0:go
to3000
2970 ifa$=chr$(134)andnl=0thensp=0:
ze=16:gosub3090:print"nein":nl=1
2980 ifa$=chr$(135)andnd=1thensp=0:
ze=19:gosub3090:print"ja  ":nd=0:go
to3000
2990 ifa$=chr$(135)andnd=0thensp=0:
ze=19:gosub3090:print"nein":nd=1
3000 goto2900
3010 rem ========================
3020 rem = programm beenden     =
3030 rem ========================
3040 printch$"E N D E !":sys64738
3050 rem ========================
3060 rem verschiedene unterpro-  =
3070 rem gramme,routinen,abfragen=
3080 rem ========================
3090 poke211,sp:poke214,ze:sys58640
:return
3100 gosub3090:o=o+1:printl$:sp=3:g
osub3090:printzt(o):o=o+1
3110 sp=10:gosub3090:printzt(o):o=o
+1
3120 sp=18:gosub3090:printzt(o):o=o
+1:sp=26:gosub3090:printzt(o):retur
n
3130 ifle=1andau=1thenb=1:d=5:s=9:t
3=1:return
3140 ifle=2andau=1thenb=13:d=17:s=2
1:t3=2:return
3150 ifle=3andau=1thenb=25:d=29:s=3
3:t3=3:return
```

```
3160 ifle=4andau=1thenb=37:d=41:s=4
5:t3=4:return
3170 ifle=1andau=2thenb=2:d=6:s=10:
t3=5:return
3180 ifle=2andau=2thenb=14:d=18:s=2
2:t3=6:return
3190 ifle=3andau=2thenb=26:d=30:s=3
4:t3=7:return
3200 ifle=4andau=2thenb=38:d=42:s=4
6:t3=8:return
3210 ifle=1andau=3thenb=3:d=7:s=11:
t3=9:return
3220 ifle=2andau=3thenb=15:d=19:s=2
3:t3=10:return
3230 ifle=3andau=3thenb=27:d=31:s=3
5:t3=11:return
3240 ifle=4andau=3thenb=39:d=43:s=4
7:t3=12:return
3250 ifle=1andau=4thenb=4:d=8:s=12:
t3=13:return
3260 ifle=2andau=4thenb=16:d=20:s=2
4:t3=14:return
3270 ifle=3andau=4thenb=28:d=32:s=3
6:t3=15:return
3280 ifle=4andau=4thenb=40:d=44:s=4
8:t3=16:return
3290 ifzt(b)=0thenzt(b)=ti
3300 ifzt(s)=0thenzt(s)=ti:return
3310 ifzt(b)>tithenzt(b)=ti:return
3320 ifzt(s)<tithenzt(s)=ti:return
3330 return
3340 rem mathetrainer ==========c64
3350 rem 38911 bytes memory     ===
3360 rem 10968 bytes program    ===
3370 rem 00056 bytes variables  ===
3380 rem 00252 bytes arrays     ===
3390 rem 00015 bytes strings    ===
3400 rem 27620 bytes free       ===
3410 rem ==========================
```

## Minotaurus

```
20 rem  minotaurus ================
30 rem (p) 03/86 commodore welt   =
40 rem ===========================
50 rem (c) 02/86 r. schmid-fabian =
60 rem c-64                       =
70 rem ===========================

80 cf$=chr$(144)+chr$(147)
81 d3$=chr$(17)+chr$(17)+chr$(17)
82 u3$=chr$(145)+chr$(145)+chr$(145
)
83 print cf$:gosub170:poke53280,11:
poke53281,11
85 fori=1to880:x=int(rnd(1)*2):prin
tchr$(164+x);:next:rem labyrinth
90 gosub600
110 ti$="000000"
120 gosub230:geta$:ifa$=""then 120
130 printa$;:goto 120
140 rem***************************
150 rem spritecursor in 11*64 ***
160 rem***************************
170 f=0:v=53248:pokev+21,1:pokev+39
,f:poke2040,11:rem f=farbe
180 a=11*64:fori=atoa+62:pokei,0:ne
xt
190 for i=a+3 to a+15 step 3:pokei,
30 :next:return
200 rem***************************
210 rem spritecursor positionieren
220 rem***************************
230 a=(peek(209)+256*peek(210)-1024
)/40:b=peek(211)
240 a=a+int(b/40):b=b+(b>=40)*40
245 if a=24 then run:rem bei aufgab
e
246 ifa=0then print cf$"sieg":print
"zeit="ti/60"sekunden":printd3$"run
"u3$:end
250 xn=24+b*8:yn=50+a*8:if xn=xa th
en if yn=ya then return
255 pokev+31,0:rem kollisionsflagge
 loeschen
260 x=(xn+xa)/2:y=(yn+ya)/2:gosub 5
00:for t=1 to 50:next t
270 if peek(v+31)>0 then gosub550:r
eturn:rem kollision mit hintergrund
300 x=xn:xa=x:y=yn:ya=y:gosub 500:r
eturn
490 rem * sprite nach x,y ******
500 pokev,xand255:pokev+16,x/256:po
kev+1,y:return
540 rem * wackeln + geraeusch *****
550 for g=1 to 70:poke54296,rnd(1)*
15:poke53270,rnd(1)*15:next
560 poke54296,0:poke53270,200
600 x=24:y=242:gosub500:xa=24:ya=24
2
690 rem  cursor positionieren ****
700 poke214,23:poke211,0:sys 58640:
return
```

# BÖRSE

Super C 116 mit 64k-RAM, 60671 Bytes frei oder Grafik, 48381 Bytes frei für nur 240,– DM.
Otto Schmälzle, Franz-Schubert-Str. 10, 7602 Oberkirch
Tel: 07802/6026 nach 20 h

---

Suche Programme für Commodore 8032 Floppy 8050, Tel: 04825/488

---

Spitzen C 64-Software über 1300 Action-Adventures und Anwenderprogr. geg. Unkosten abzugeben. Liste mit Preisangabe kostenlos anfordern. A. Wittgen, Postlagernd 073861 C, 5060 Bergisch Gladbach

---

Verkaufe: CP-80X + Kabel + Handbücher + Hardcopypgr. direkt an C 64 + C 128 Grafik 1280 Punkte/Zeile, 4 Mon. alt! Wie neu! VB 750,– DM. Näheres unter: 06106/75479, Volker Bliß, Frankfurterstr. 106, 6054 Rodgau 3

---

Für VC 20: Verkaufe einige 24k-RAMS. Voll schaltbar: 0-3-8ß16-24k. Auch versch. Adressbereiche schaltbar. Info geg. Rückporto! Friedel Heinz, An der Halde 43, 4330 Mülheim a.d. Ruhr

---

5 1/4 Zoll / 3 1/2 Zoll Disketten zu Spottpreisen
Rolf Küttel
Sportplatzweg 4
6440 Brunnen
Te: 043/311069

---

VC 20 VC 20 VC 20 VC 20
Videospiele für Grundversion bis 8k, Angebotsliste gegen frankierten Rückumschlag Andreas Drescher, Harrier Str. 29, 2880 Brake

---

VC 20 VC 20 VC 20 VC 20
Verkaufe 16kROM + Modulbox + Schach für nur 120,– DM!!! Ralf Hopperdietzel, Th.-Storm-Str. 22, 8670 Hof

---

Der C 64/VC 20 mini-club sucht immer wieder neue Mitglieder (auch PC 128er)! Info erhältlich gegen Rückporto! Außerdem suchen wir noch Kontakte mit anderen Computer-Clubs!!! Meldet euch bitte bei uns!!! Dirk Smits, Pf. 1314, 415 Krefeld 29

---

VC 20-Freaks aufgepasst!
Tausche Supersoftware (z.B. Space Attack, Abductor, Galaxions, 3 D-Pac Man, Star Wars, Scramble, Tacco, u.s.w) Schickt eure Listen an: Frank Ludwig, Keltenstr. 48, 6630 Saarlouis-Roden

---

C 16 und C 116
„60671 BYTES FREE“, RAM für riesige Programme, Software. Info bei: Dipl.-Ing. H. Stechmann, Pf. 210, 2152 Horneburg.

---

CBM 4040 Doppelfloppy zu verkaufen! VB 1400,– DM. Tel: 07731/52586 ab 18.00 Uhr

---

C 128 Suche Software! Auch für CP/M Modus. Bitte schreibt an: Jürgen Tetzel, Grellestr. 28, 2840 Diepholz, Antwort garantiert!

---

Zu verk.: C 64 + 1541 + Formel 64 + Toutsch-Tublet + GP 500 VC + S/W-Fernseher + 1 Game-Mod. 3 2 Spielecas + 100 besp. Disketten + Staubschutz + 2 Joysticks + 3 14 C 64 Fachbücher + Comp.Zeitsch. + Diskettenbox. Alles kompl. f. 2300,– DM. Werner Lorber, Postfach 1951, 8600 Bamberg

---

Verkaufe original Startexter 30,– DM, Stardatei 30,– DM, Disketten DD, DS, 10 St. 15,– DM, Centronics-Drukker-Mabel an User-Port 20,– DM. Softwaretausch, Tel: ab 18 Uhr, 069/782718

---

Original Kalkumat, Datamat, Textomat plus, Profi Painter zu verkaufen. E. Baudach, Tel: 0221/8902588

---

Versende VC 20 Spielkaseten gegen 5,– DM in Briefmarken o. Geldschein. Ted Schiwietz, Eichenring 28, 8059 Neufinsing, Tel: 08121 /7445.
Brauche Hilfe bei Benutzung von Eprombrenner im 128er Modus.

---

Partner(in) gesucht zum Computern: Vielleicht nur als intensives Hobby, besser: Sich damit selbstständig machen (Grundlagen bei mir vorhanden). Erfahrungen: CBM 8032, C 64, CANON AS-100 (CP/M, MS DO5, Olivetti M 20.
Dr. Pabst, 06171/24826

---

Suche gut erhaltenen, preiswerten Drucker und Anwenderprogramm für VC 20. Angebote an: Marcus Hauer, Scheffelweg 28, 6972 Tauberbischofsheim

---

Verkaufe meine C 16/ C 116/ plus 4-Spiele zu Schleuderpreisen. Kostenlose Liste von: Udo Hölzer, Erzstr. 15, 4630 Bochum 1

---

Suche Gleichgesinnte zwecks Erfahrungsaustausch in meiner Gegend (Michelstadt - Beerfelden - Hirschkorn etc.) Besitze Commodore 128. Bei Interesse ruft bitte an. Tel: 06068/3456

---

Verkaufe Datasette mit 7 Inputkassetten von 6/85 bis 3/86, sowie 3 Spiele (Superpipeline, Bonka, BMX Training) für 170,– DM, NP. 230,– DM.
M. Schröder, Planstr. 13, 6450 Hanau 6, Tel: 06181/ 73001

---

Suche gebrauchte Floppy VC 1541 für C 64 (unter 300,– DM). Suche Infomatrial und technische Beschreibungen über Erweiterungsmodule für C 64. Angabe Preis/Bezugsquelle an: Günter Pauli, Adam Karrillonstr. 1-3, 65 Mainz

---

VC 20 + 8k
MC-Monitor (Assembler, Disas, Find, Dump, Trace usw.) + Fasttape C = 10,– DM, D = 15,– DM, desgl. m. Diskmonitor C = 15,– DM, D = 20,– DM inkl. Porto. Vorauskasse oder NN. T. Grünner, Mark-Twain-Str. 5, 3300 Braunschweig, Tel: 0531/314264, PS: Auch Tausch gegen Modulprogramme.

---

C 128/ 640 x 200 Grafik. Zeichnen sie auf C 64 Grafikprogramme. HIG 128 überträgt Bilder und koppelt sie. Mit Anl. und Disk 50,– DM. M. Uetzels, Annabergstr. 1a 4040 Neuss 22, Tel: 467441

---

Suche Software jeglicher Art für C 16 (Cas). Angebote mit Preisen an: Peter Höfert, Kiefernweg 27, 4512 Wallenhorst 1

---

Die neueste Software für ihren C 64 gegen Unkosten abzugeben. Über 1300 Pr. Liste kostenlos bei: A. Wittgen, Postlagernd 073861, 5060 Bergisch Gladbach 1

---

Verkaufe und tausche Programme für C 16. Christian Schmidt, Hauptstr. 62, 8737 Sulzthal

---

Verk. für C 16/ 116/ plus 4 Daley Tompsons Star Events. Ein Sportspiel mit 7 Disziplinen u. wirklich hervorragender Grafik. Es kostet inkl. Kassette oder Disk u. Anl. 15,– DM u. ist geg. Barzahlung zu beziehen bei M. Mehlau, Schönhardtweg 9, 7165 Fichtenberg

---

!!!Achtung!!!
Ich tausche und verkaufe laufend die neuesten und besten VC 20-Games! Meldet euch bei: Thomas Kneuper, Pfarrer-Jansen-Str. 7, 4418 Nordwalde, Tel: 02573/3478

---

VC 20
Suche und tausche Software für VC 20. Interessiert besonders an neuerer Software. Marcus Timpel, Friesenweg 23, 5090 Leverkusen, Tel: 02171/44648

---

Suche VC 1541 Floppy für C 64. Angebote an: Holger Stutzke, Am Brückentor 37, 4018 Langenfeld

---

Suche Drucker für VC 20, z.B. GP-100 VC, VC 1525, VC 1526 oder anderen mit mindestens 7 x 7 Punktmatrix für maximal 250,– DM. Tausche auch Software. Raimund Paul, Stuttgarter Str. 44, 1000 Berlin 44, Tel: 030/ 6869231

---

C 64 C 64 C 64 C64 C64
Wir (3 Schüler) übernehmen ihre Programmierarbeit!!! Spottbillig!!!
Info: bei Stefan, Tel: 0711/ 6873162

---

Verk. VC 20 (11 Mon.) + Datasette + Handbuch + Schachmodul Sargon + Cassetten (Basic, Earth Defense, UFO, Black Max, Spielautomat 17 + 4), um 280,– DM. Alois Unverdorben, Webersdorf 16, A-5132 Geretsberg Tel: 0043/7748/7110 ab 17 Uhr

---

C 64 C 64 C 64 C 64
Verkaufe C 64 Programm (über 400 Spiele)
Daniel Suter
Hauptstr. 31
CH-932 Arbon
Tel: 071/467924

---

Dringend!!!
Schüler sucht SX 64. Nur vollständig in Ordnung. Kann bis 250,– DM zahlen. Angebote an: Ulrich Müller, Hertzstr. 45, 8600 Bamberg

---

C 64. Suche Tauschpartner. Kontakte zu 64er-Besitzern, auch Clubs, zwecks Erfahrungsaustausch rund um den 64er. Tel: 07448/1032

---

!!!SUCHE!!!
Schüler sucht kostenlos defekten Computer und Zubehör zu Bastelzwecken. Ulrich Müller, Hertzstr. 45, 8600 Bamberg

---

**Anwendung der DTRL-Taste**

Alles, was man mit PRINT CHR$(x) mit x<32 erreichen kann, wird auch ausgeführt, wenn zusammen mit der CTRL-Taste ein Zeichen mit Bildschirmcode <32 gedrückt wird. (Handbuch S. 133)

---

**C64** — **Programmiertricks**

HIRESGRAPHIK (Beispielprogramm).
Wie man eine hochauflösende Grpahik programmieren kann, zeigt dieses Beispiel:

```
95  rem ************** bitmap schützen
100 poke51,255:poke52,31:poke55,255:
    poke 56,31
105 def fn y (x) = 50*sin(X/30)+100
110 rem ***************** hires ein
115 poke53272,peek(53272)or8
120 poke53265,peek(53265)or32
125 rem ***************** farbe wählen
```

---

**C64+VC20** — **Tips & Tricks**

Wichtige Abkürzungen

(im Kleinschriftmodus eingeben)

| | | | | |
|---|---|---|---|---|
| ? | PRINT | aber! | pR | PRINT# |
| pO | POKE | | pE | PEEK |
| vE | VERITY | | CH | CHR$ |
| sA | SAVE | | lI | LIST |
| sT | STOP | | stE | STEP |
| rI | RIGHT$ | | leF | LEFT$ |
| cL | CLR | | c1O | CLOSE |
| gO | GOTO | | goS | GOSUB |
| iN | INPUT# | (INPUT hat keine) | | |
| sP | SPC( | ! | (Klammer) | |

---

**C64** — **Programmiertricks**

SPRITES (Beispielprogramm)
Will man Sprites in Programmen verwenden, so muß man folgendes beachten: V = 53248
Es gibt 8 Sprites (n=) bis 7)
1 Sprite belegt 63 Bytes
Adresse für Sprite n steht in 2040+n
(11 bedeutet: 11*64=704)
Farbe für Sprite n in V+39+n
Sprite n einschalten:
POKEV+21,PEEK(V+21) OR 2?n
Sprite n auf x,y setzen:
POKEV+2*n,x:POKEV+2*n+1,y

---

**C64** — **zum Handbuch (CHR$)**

einige CHR$Codes sind nicht im Handbuch aufgeführt:

| | |
|---|---|
| OHR$(129) | =Orange |
| CHR$(149) | =Braun |
| CHR$(150) | =Hellrot |
| CHR$(151) | =Grau 1 (dunkel) |
| CHR$(152) | =Grau 2 (mittel) |
| CHR$(153) | =Hellgrün |
| CHR$(154) | =Hellblau (normal) |
| CHR$(155) | =Grau 3 (hell) |

---

**C64** — **Programmiertricks**

FUNKTIONSTASTEN (Beispielprogramm)
Die Funktionstasten haben die ASCII-Codes 133-140. Wie man sie in Basic verwenden kann zeigt dies kleine Programm. Es verschiebt Sprite 0 (vorher einschalten!)

```
190 v=53248:rem ************* start vic
230 rem *********** funktionstasten lesen
235 get a$:if a <133 or a>140then235
250 on a-132 goto 260,270,280,295,310,
    365,395,410
```

---

**C64+VC20** — **POKE 788,x**

STOP-Taste

x=52 ausgeschaltet
x=49 eingeschaltet

für VC20

x=194 ausgeschaltet
x=191 eingeschaltet

---

**C64+VC20** — **Programmiertricks**

MINITEXTPROGRAMM (CBM 5/85)
(Groß/Kleinschrift umsch.):
Im Direktmodus eingeben:

```
pO44,10:pO10*256,O:NEW (RETURN)
s=2048:fOi=1to85:pO,O:pOs+1,1:
pOs+2,2:pOs+3,i:pOs+4,O:pOs+5,34:
s=s+6:nE:pOs-4,0      (RETURN)
pO44,8:SYS42291:1I-22
```

(VC20:SYS50483,statt 2048 den Einschaltwert von PRINT pE(44)*256
Jetzt können Sie alles schreiben.
Groß/Klein, mit Cursorsteuerung ...
Jede Zeile mit (RETURN) abschließen.

```
130 rem ** zf=zeichenfarbe hf=hintergr.
135 zf=0:hf=6:f=16*zf+hf
140 for i=1024 to 2023:poke i,f:next
145 rem *************** bitmap löschen
150 b=8192:fori=Oto7999:poke b+i,0:next
155 rem **************** Kurve zeichnen
160 for x=Oto319:y=fn Y(x):gosub175
165 next:goto190
170 rem **************** punkt setzen
175 be=(xand504)+40*(Yand248)+(yand7)
180 bi=7-(xand7)
185 pokeb+by,peek(b+by)or(2 bi):return
190 poke 198,0:wait198,1:poke 198,0
620 sys65409:rem ******* hires aus
```

Folgende Kombinationen sind sinnvoll:

CTRL + 'H' : blockiert SHIFT C=
CTRL + 'I' : entrieg. SHIFT C=
CTRL + 'M' : RETURN
CTRL + 'N' : Kleinbuchstaben
CTRL + 'Q' : CURSOR runter
CTRL + 'R' : REVERS ON
CTRL + 'S' : HOME
CTRL + 'T' : DELETE
CTRL + ':' : CURSOR rechts

Drückt man z.B. statt RETURN CTRL/M, so wird die Ausführung (wegen CTRL) verlangsamt.

Beispielprogramm (Sprite 0):

```
190 v=53248:rem ************* start vic
200 rem ****************** sprite 0 ein
205 pokev+21,1
210 ren ******* sprite 0 auf x,y setzen
215 x=160:y=120: poke v,x:poke v+l,y
440 poke 2040,11:rem sprite 0 in 11*64
500 rem ***** spritespeicher löschen
510 fori=704to767:pokei,0:next
525 rem ********** spritefarbe hellblau
530 for i=39 to 46:poke v+i,14:next i
535 rem ****** daten fadenkreuz ******
540 for i=0 to 7:read i:poke 704+i*3,d:          next
545 data 16,16,16,238,16,16,16,0
```

Bemrkung:

Die meisten Befehle können durch die ersten zwei Buchstaben abgekürzt werden, wobei der zweite zusammen mit der SHIFT-Taste gedrückt werden muß.

```
255 rem ************** sprite 0 rauf
260 poke v+1,peek(v+1)-1:goto 235
263 rem ************** sprite 0 runter
270 poke v+1,peek(v+1)+1:goto 235
275 rem ************** sprite 0 rechts
280 if peek(v)=255 then poke v,0:
    poke v, peek(v)+1:goto 235
290 rem ************** sprite 0 links
295 if peek(v)=0 then poke v,255:
    pokev+16,peek(v+16and254:goto235
300 poke v,peek(v)-1:goto 235
```

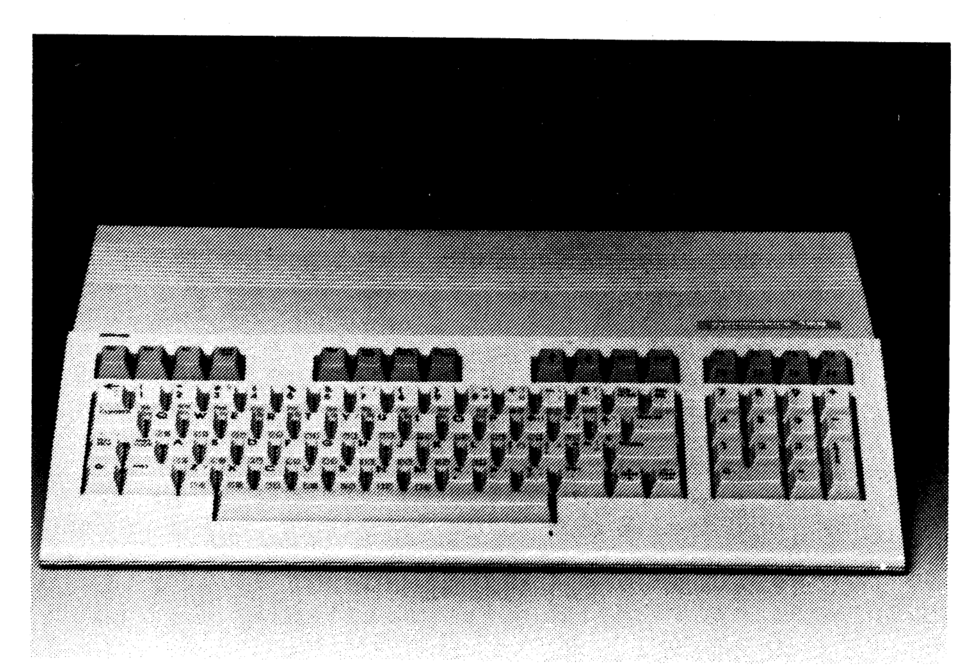
Wie Basicprogramm abspeichern oder als Listing ausdrucken:

OPEN1,4:CMDI:LIST (RETURN)
PRINT#1:CLOSE1 (RETURN)

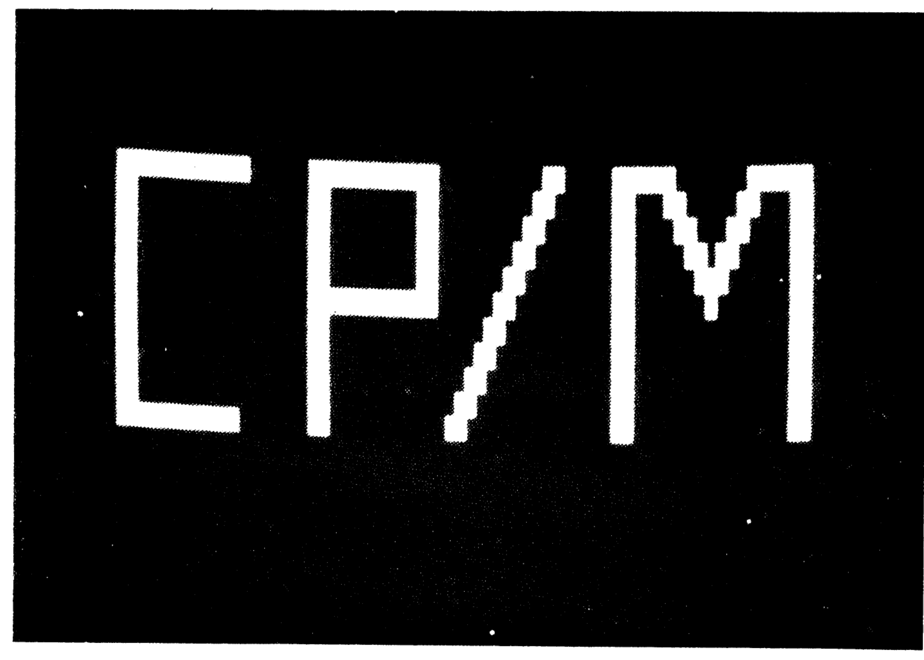
Man kann auch verhindern, daß die Zeilennummern ausgedruckt werden:
vor LIST: POKE 22,35 (normal:,25)
Doppelten Zeilenabstand erreicht man mit:
OPEN128,4:CMD128:LIST
Die Anführungszeichen kann man durch Einlegen in 2. Papier mit 1 cm Falz auf den diese gedruckt werden verhind.

Anwendung:

Am Anfang des Programms muß POKE 788,52 stehen, dann kann das Programm, nachdem es diese Stelle erreicht hat, nicht mehr mit der STOP-Taste abgebrochen werden.
Für x können auch andere Werte ausprobiert werden.



# 128 + CP/M




**Unsere Serie über CP/M hat sehr viel Anklang bei den Lesern gefunden. Das zeigt uns, daß sich doch noch viele Anwender ernsthaft mit diesem doch schon etwas betagten Betriebssystem beschäftigen wollen. Hier nun, wie versprochen, der zweite Teil der Serie.**

Sicher haben Sie sich bei Ihrer Arbeit mit dem C128 schon oft gefragt, wie man die Funktionstasten mit Strings belegen kann. Unter CP/M existiert dafür der sog. String-Editor, mit der Tastenkombination

CTRL-SHIFT-CURSOR RECHTSTASTE

Und das alles bitte gleichzeitig (böse Zungen in der Redaktion behaupten, der Redakteur hätte seitdem eine Geierkralle an der linken Hand). Auf dem Bildschirm erscheint der aktuelle Inhalt der Taste auf dem Schirm.

Zusammen mit der CTRL und rechten SHIFT-Taste stehen Ihnen jetzt die folgenden Editierkommandos zur Verfügung:

| | |
|---|---|
| o ——〉 | : Cursor rechts |
| o 〈—— | : Cursor links |
| o — | : Zeichen unter Cursor löschen |
| o + | : Zeichen einfügen |
| c CR | : Abschluß der Eingabe |

Ein Tip am Rande: Seien Sie vorsichtig mit dem Editieren, es ist sehr leicht passiert, daß man einzelne Tasten nicht anschlägt, und somit Funktionen ausführt, die man gar nicht haben will. Sicher haben Sie jetzt auch schon bemerkt, daß die Strings, die auf den Funktionstasten liegen, auf der eben beschriebenen Weise erzeugt wurden. Tabelle 1 zeigt die komplette Belegung der Funktionstasten.

Leider ist diese Art des Arbeitens sehr fehlerträchtig. Um uns die Arbeit zu erleichtern, existiert das Programm „KEYFIG", die Abkürzung für KEY KONFIURATION.

Mit „KEYFIG" kann man fast alle Tasten des 128'er neu belegen. Die Ausnahmen sind:
- die SHIFT-Tasten
- die SHIFT-LOCK Taste
- die CONTROL-Taste
- die CBM-Taste.

Um Ihnen die Arbeit zu erleichtern, sind erklärende Menüs und alle verfügbaren Optionen bei jedem Schritt vorhanden. Sie können mit den Cursor-Tasten und der RETURN-Taste beliebig auswählen. Sollten Sie das Programm vorzeitig verlassen wollen, mit Ctrl-C geht dies, ohne daß Daten auf die Diskette geschrieben werden. Aufgerufen wird das Programm in der CP/M üblichen Weise mit

A〉KEYFIG

Als gewissenhafter Anwender beantworten Sie die Frage nach der angebotenen Hilfe natürlich mit Y für „YES". Für die noch nicht Besitzer zeigt Bild 1 die möglichen Punkte, für die das System seine Hilfe anbietet. Mit der ersten Option (——〉done help〈——) kommen Sie übrigens wieder aus dem Help-Modus heraus. Sie können mit dem folgenden Menü die zu bearbeitende Tastaturbelegung auswählen: Die englischen Texte bedeuten dabei folgendes:
- Standard Tastaturbelegung
- Tastaturbelegung von der CP/M Boot Disk holen
- Aktuelle Tastaturbelegung
- Ihre letzte Arbeitsdatei

Mit der Option „Edit a key definition“ werden anschließend die neuen Strings eingegeben. Auch hier nimmt Ihnen ein Menü wieder die meiste Arbeit ab. Belassen Sie es für den Anfang mit der Umbenennung der „Z“ und „Y“ Tasten. Anschließend wird Ihre Arbeit auf Diskette abgespeichert.

Auch hier gilt wieder der eiserne Grundsatz, daß Sie nie mit der Originaldiskette arbeiten sollten, sondern immer mit einer Sicherheitskopie.

**ZEIGT SICH: SHOW**

Ein ebenfalls hilfreiches Programm ist SHOW, das Sie auf der Utility-Diskette finden. Es zeigt die möglichen Optionen einer Diskette an.

Auch hier gibt es interessante Optionen, die einfach mit

**A>SHOW (OPTION)**

aufgerufen werden. Sehr nützlich sind

Zeilen-Beschränkung auf dem Bildschirm. Sie können jederzeit mit Ctrl-S anhalten, oder mit Ctrl-C stoppen. Übrigens können Sie so auch Source-Files in jeder beliebigen Sprache ansehen. Mit

**A>ERATOS.PAS**

beispielsweise listen Sie ein Turbo-Pascal Source-File auf. Natürlich können Sie es mit Ctrl-P auch auf den Drucker umleiten.

```
keyfig
C128 SOFT KEYBOARD PROGRAM
        3 June 1985Welcome to the Commodore C128 Keyboard
Definition program.  Do you want help? Help  is  available  on  the
following topics:
--> done help <--
--> General Usage <--
--> Setting up your work file <--
--> What to do with your work file <--
--> Key values <--
--> Selecting a key to edit <--
--> Logical/Physical Colors <--
--> Editing keys <--
--> Assigning/Editing Strings <--
--> Assigning Colors <--
--> Assigning Special Functions <--
--> Assigning HEX values <--
--> Finishing up <--
--> For experts only <--
Use the up and down arrow keys to scroll
through the menu; type the return key to
select the topic on which you want help.Are you sure?
```

Hilfsfunktionen

**Auch unter CP/M: RENAME**

CP/M Plus bietet Ihnen die Möglichkeit, Dateien wie in Basic umzubenennen. die Syntax lautet ganz einfach:

**RENAME Neuername – Altername**

Natürlich kann auch dieser Befehl mit REN abgekürzt werden. Um gleich mehrere Dateien zu verändern, sind auch hier die Wildcarts (oder Joker, wie sie wollen) „*“ und „?“ erlaubt. Beispiel:

**A>REN S*.TXT=A*.TXT**

Wenn Sie ganz sicher gehen wollen, geben Sie nur **RENAME** ein, das System fragt Sie nach neuem und altem Namen.

| Option | Bedeutung |
|---|---|
| Space | Freien Speicherplatz und Zugriffsmodus (RW oder RO) anzeigen. |
| Drive | Anzeige der Laufwerksparameter. |
| Users | Anzeige der benutzten Userbereiche. |
| Dir | Freie Directory Plätze (nicht verwechseln mit freiem Speicherplatz). |

**OHNE UMWEGE: TYPE**

Mit dem residenten Kommando TYPE ist es möglich, Dateien direkt auf dem Bildschirm auszugeben. Sie müssen daher nicht unbedingt ein Textsystem laden, um Ihre Texte anzuschauen. Geben Sie ein

**A>TYPE ARTIKEL.TXT (NOPAGE)**
erscheint der Text ohne die 24-

**MULTIFUNKTIONSBEFEHL: SET**

CP/M gibt Ihnen mit **SET** einen Befehl in die Hand, mit dem sehr einfach und komfortabel Paßwörter und DAtei-spezifische Parameter gesetzt werden können. Weiterhin dient **SET** zum Verändern der CP/M Fileattribute READ-WRITE (RW) und READ-ONYL (RO).

Eine der Anwendungen von SET, die sofort ins Auge sticht, ist die „Time-Stamp“ Funktion. Hier „drücken“ Sie jedem File auf der Diskette einen Zeiteintrag auf. Diese Funktion ist für die Anwender sehr hilfreich, die sehr viele Texte oder Programme schreiben. Sie hat aber einen entscheidenden Nachteil: es geht etwas Speicherplatz auf der Diskette verloren.

Diese Timestop-Funktion benötigt einige Vorarbeiten.Formatieren Sie zunächst eine Diskette. Anschließend rufen Sie das Programm INITDIR.COM von der Utility-Disk auf. Dieses Programm richtet die Diskette für die Zeiteinträge ein.

Jetzt können Sie mit dem SET-Befehl und einigen Parametern das Directory beeinflussen. Dabei kommen drei Parameter in Frage:

SET A: (CREATE=ON)
SET A: (ACCESS=ON)
SET A:(UPDATE=ON)

Die Optionen bedeuten im einzelnen:

- CREATE: Der Zeitpunkt der Erstellung der Datei wird eingetragen.
- ACCESS: Jeder Zugriff auf die Datei wird im Directory festgehalten. Eine DIR-Anweisung wird dabei nicht berücksichtigt.
- UPDATE: Bei jedem Schreibzugriff wird eine Eintragung vorgenommen.

CREATE und ACCESS-Einträge können nicht gleichzeitig eingeschaltet sein, ein Befehl

A⟩SET A: (CREATE=ON, UPDATE=ON)
ist aber ohne weiteres erlaubt.

**PASSWÖRTER UND LAUFWERKS-ATTRIBUTE**

Um Ihre Dateien vor unbefugtem Zugriff zu schützen, gibt es die PASSWORD-Option. Um ein Paßwort zu setzen, benutzen Sie

A⟩SET A:(PASSWORD=AETSCH)

Mit diesem Kommando ist ab sofort die gesamte Diskette mit einem Paßwort versehen. CP/M fragt also vor jedem Zugriff nach; merken Sie sich Ihre Paßwörter gut, nachdem die Disketten so geschützt worden sind, besteht ohne Paßwort keine Möglichkeit mehr, an die Daten heranzukommen. Ausgeschaltet wird diese Funktion mit

A⟩SET A: (PASSWORD=⟨CR⟩)

allerdings erst, wenn Sie die Frage nach dem Paßwort ordnungsgemäß beantworten. Um die Verwirrung vollständig zu machen, können Sie sogar angeben, für welche Art des Dateizugriffes das Paßwort gelten soll. Betrachten wir einmal folgende Befehle:

A⟩SET A: TEXT(PASSWORD=TURBO)

Die Datei TEXT bekommt also das Paßwort TURBO. Sie wollen aber in die Datei etwas hineinschreiben, aber sichergehen, daß sie nicht gelöscht oder umbenannt wird. Dazu geben Sie ein:

A⟩SET TEST(PROJECT=DELETE)

Mit DELETE ist die Datei gegen versehentliches Löschen oder Umbenennen geschützt. Neben DELETE sind noch die Optionen READ, WRITE und NONE möglich. Mit READ sind alle denkbaren Dateizugriffe unmöglich gemacht worden. WRITE ermöglicht zwar das Lesen, nicht jedoch den Schreibzugriff (incl. Löschen und Umbenennen), DELETE haben wir schon betrachtet und NONE schaltet den Paßwortschutz ganz einfach aus. Beachten Sie bitte, daß mit diesem Befehl keine COM Dateien geschützt werden können.

Wenn Ihnen vor lauter Optionen und Parameter schon der Kopf raucht, so kann ich Sie beruhigen: die Laufwerksattribute sind nicht sehr umfangreich. Genauer gesagt gibt es derer nur zwei, nämlich die schon erwähnten READ-ONYL und READ-WRITE.

A⟩SET A:(RO)
A⟩SET A:(RW)

RO schützt das gesamte Laufwerk vor einem Schreibzugriff, Sie setzen praktisch einen „Software-Schreibschutzaufkleber". Logischerweise funktionieren dann auch PIP oder ERASE nicht. Der Schutz wird mit dem zweiten Befehl wieder aufgehoben.

Damit schließen wir den zweiten Teil unserer CP/M Serie. Wenn Sie jetzt Lust bekommen haben, dann setzen Sie sich an den Computer und probieren die angesprochenen Dinge aus. Sie können sich dann schon als Fast-Profi bezeichnen.

Was es mit dem Programm ED. COM auf sich hat, warum es sich lohnt, das Programm SUBMIT. COM auszuprobieren und wie man sich durch eine 1150 Gramm schwere Dokumentation „durchfrißt", das alles erfahren Sie im nächsten Teil der Serie.

TABELLE 1:
Stringbelegung nach dem Booten. Links der Hexcode, rechts der String.

| | | |
|---|---|---|
| 80:„F1" | 81:„F2" | 82:„DIR⟨CR⟩" |
| 83: „DIR" | 84:„F5" | 85:„F6" |
| 86:„F7" | 87:„3 JUNE 85" | 88:„F9" |
| 89:„F10" | 8A:„F11" | 8B:„SCR ⟨" |
| 8C:„SCR ⟩" | 8D: „SCR ⟨" | 8E:„SCR ⟩" |
| 8F:„F16" | | |

90 bis 9E: „F17" bis „F31"
9F: „HELP"

## Sind Sie ein Computerfreak?

0 – 14 Punkte:
Entweder haben Sie Ihren Computer noch nicht allzu lange oder aber Sie sind einer der bewundernswerten Menschen, die von ihrer EDV-Anlage noch relativ unabhängig sind.

15 – 25 Punkte:
Aha!
Erste Anzeichen dafür, daß Sie ein echter Computerfreak werden; sind schon vorhanden.
Nicht mehr lange und Sie erreichen bei diesem Test die VOLLE Punktzahl.

26 – 30 Punkte:
Sie sind ein echter Freak.
Ohne Ihren Computer können Sie nicht mehr leben. Wahrscheinlich hat Sie Ihre Frau/Freundin schon verlassen oder sie hat es zumindest schon mehrfach angedroht.
Ein Wunder eigentlich, daß Sie Ihre Zeit für so dumme Dinge wie diesen Test verschwenden, zumal wir hier in Deutsch und nicht in ASCII-Zahlencode schreiben.

# BÖRSE

Verkaufe: 1 Graupner Elektromotorrad NP 200,— DM für nur 140,— DM oder tausche gegen Joystick Comp. Pro II + Lightpen für C 64
G. Reitzer, Blumenstr. 14, 7258 Heimsheim

---

Achtung! Dringend preisgünstige Spiele nur auf Disk zu kaufen gesucht! Spiele für C 64 und C 128.
An: Richard Gutermuth
Eichenweg 1
5340 Bad Honnef/Rhein 1
Tel: 02224/3445

---

Suche, tausche, verkaufe alle mögliche VC-20 Software. Große Auswahl an Progr. vorhanden. Schickt Listen an: Christoph Krohn, Goethestr. 35 a, 5980 Werdohl

---

Suche alle Programme für 8032 SK. Tel: 09526/1680

---

Verkaufe Commodore 16 + Datasette + Anschlußkabel + Adapter + Handbuch + Modul + 2 Spiele!!! (Technisch + optisch o.k.)
Markus Stuettgen
Zaunswinkelstr. 42
4019 Monheim Preis: 150,—
Tel: 02173/56325

---

VC20 VC20 VC20 VC20
Verkaufe VC-20 + Data-Sette + 32k + Joystick + Literatur (Listings, Bücher) + Software + Schachmodul: VB 400,— DM, Thomas Reinhold, Hangweg 8, 7102 Gellmersbach, Tel: 07134/8229

---

C-128 C-128 C-128 C-128
Einsteiger sucht C-128-User zwecks Information, Erfahrung- und Software-Austausch. Heiko Hamann, Timmersgarten 24, Datteln 3

---

SUCHE SUCHE SUCHE
Für C-128 das RP-System Preisvorstellung: 20,— bis 25,— DM. Disk mit Handbuch. Bitte schriftliche Angebote an: Axel Fetzner, Kurpromenade 3, 7506 Bad Herrenalb

---

Verkaufe Warneke's Mailbox Telefonbuch für 6,— DM inkl. Porto und Verpackung.
Ulrich Müller, Hertzstr. 45, 8600 Bamberg

---

Ärger!!! mit Handbüchern, Programmbeschreibungen u. Anleitungen in Englisch? Ich übersetze ins Deutsche, schnell und korrekt.
Udo Hofmann, Im Buschfeld 21, 5482 Grafschaft 3, nur werktags am Vormittag tel. 02641/4160

---

VC-20 Tausch und Verkauf von Programmen. Verkaufe Pokermodul Fr 20,—; Originalkassetten Commodore, Autorallye, Earth Defense, zu je Fr 10,—. Suche Drucker zu VC 20 und Flugsimulator. Adi Kurzen, Arnika, CH-3818 Grindelwald

---

VC-20-Fan sucht schrottreifes Zubehör. Tausche Software (Anwendeprogr./Spiele) GV-16k — nur Cas — Liste an: Frank Thiel, Hauptstr. 54 5231 Mudenbach, Tel: 02688 /633

---

Achtung! Original D-Base II, Multiplan, Wordstar für Commodore 128 günstig zu verkaufen. Suche Lösungen für das Installateurhandwerk für C 64 und Atari 520 ST., schreibt an C. Radlgruber, postlagernd, 8399 Neuhaus am Inn

---

Achtung! Suche C-64, Floppy, Drucker. Kaufe C-64, Epson, Floppy. Suche Drucker, C-64, Floppy. Kaufe Floppy, Drucker, C-64, schreibt an C. Radlgruber, postlagernd, D-8399 Neuhaus am Inn

---

Viel Service für wenig Geld bietet der TI-CLUB BAUNATAL seinen Mitgliedern. Info gegen 50Pfg. oder Clubheft gegen 3,— DM anfordern TCB, Matthias Orf, Birkenallee 34, 3507 Baunatal 1, Tel: 0561/497990

---

Drucke Ihre Programme für nur 5,— DM je Programm. Info 50Pfg. anfordern oder Datenträger + Geld an Matthias Orf, Birkenallee 34, 3507 Baunatal 1

---

Viel Maschinensprache für 32k + Cassette gibts jetzt. Alles sehr preiswert. Info 1,— DM — anfordern, es lohnt!!! Matthias Orf, Birkenallee 34, 3507 Baunatal 1

---

Der TI-CLUB BAUNATAL hat Einkaufsvergünstigungen für seine Mitglieder. Haben Sie auch welche? Wenn nein, dann Info 50Pfg. oder Clubheft 3,— DM anfordern bei: TCB, Matthias Orf, Birkenallee 34, 3507 Baunatal 1, Tel: 0561/497990

---

Verkaufe für C-64 Ascom-Akustikkoppler mit Software für 180,— DM.
Suche Software für C-128. (Keine Raubkopien)
Wolfgang Hewig, Postdamm 18, 4460 Nordhorn, Tel: 05921-5019

---

Suche Floppy 1541 oder Epson-Drucker mit Görliz Interface sowie Eprom-Programmiergerät im Tausch gegen VHFIUHF Computerscanner 32000 Frequenzen u. digitale Anzeige. AM + FM. Angebote an: K. Kohler, Ulmenweg 6, 7074 Mögglingen

---

VC-20
Suche preiswerkte 8 - 16k Erweiterung + Mathe. Programm für Kurvendiskussion. Willi Schulte-Werflinghoff, Herderstr. 3, 4220 Dinslaken Tel: 02134/54271

---

Suche Drucker MPS 801
Wer tauscht MPS 801 gegen C 16 + Datasette + Joystick + Programme???
Drucker sollte in Ordnung sein! Christian Schmalzgruber, Hoefter Str. 20, A-5280 Braunau/Austria
Tel: 07722/48852

---

Ich suche Programme für den COMMODORE VC 20, Spiele und 16 KBY ERW. Bin 12 J. alt.
Michael Bannöhr, Wipperteich 40, 3180 Wolfsburg 11, Tel: 05363/2395

---

VC 20 Suche dringend Modulbox 3 - 6 Steckplätze, 8k Speichererweiterung, 16k Speichererweiterung, 40/80 Zeichen Karte, Software zur DFÜ u. Kass. Angebote an: Siegfried Laas, Linienstr. 28, 4000 Düsseldorf 1, Tel: 0211 /722165

---

C 16/ 116: Suche gebr. 16k-Erweiterung - außerdem suche ich jede Menge Tips + Tricks für den C 16! Suche noch Kopierprogramm und andere Software für den C 16. Tausche und verkaufe auch selbst Programme! Meldet euch bei: Andreas Kleinemeier, Hoelderlinstr. 19, 4837 Verl 1

---

Achtung! Suche Tauschpartner für C 64 + C 128 + CPM Anwendungsprogramm: Listen zusenden — 100%ige Antwort. Auch DFÜ Freaks meldet euch. Tel: 02404/ 61765. Verkaufe Newroom Bilderkatalog, Ausdruck 10,— DM.

---

Verkaufe Programme für C 16 Dateien, Spiele u.a. Jede Datei 400 Sätze — auch Tausch! Suche für C 16 das Spiel Hero. Melden bei: Christian Schmidt, Hauptstr. 62, 8737 Sulzthal.
Suche Datasettenadapter für C 64.

---

Verkaufe VC-20 119,— DM; Datasette 45,— DM; 2 Modul-Spiele: STAR BATTLE + Räuber und Gendarm a 18,— DM. Bei kompl. Abnahme gratis 6 Originalkassetten dazu. Tel: 09732/4297.

---

Für C 16 — Plus 4
Suche 20 Stecker für Joyans. Interface User/V 24, User RS 232, V 24 / RS 232. Akkustkoppler mit Interface + Treiber, Plus 4 sowie Hardware-Unterlagen gegen — Bares — Uwe Peters, Tannenweg 9, 2351 Trappenkamp, Tel: 04323/3991

---

Commodore-Club-Dixy!
Infos gegen Rückporto bei
Dieter Plaum
Galileistr. 9
6800 Mannheim 1
DIXY-CLUB-COMMODORE

---

VC 20 VC 20 VC 20 VC20
16k ROM, Orig. Kassetten, Software und orig. Module — alles billig abzugeben.
Markus Hauer, Hindenburgstr. 5, 5090 Leverkusen 1, Tel: 0214/47780

---

Suche 1541! Zahle bis 200,— DM. Tel: 09552/1432 (nur nach 14.00 Uhr)

---

SOS SOS SOS SOS SOS
Suche dringend Software für Commodore VC 20 + 16k. Nur Briefe mit Rückporto werden beantwortet. Angebote an: Boris Beutler, Düppelstr. 31, 4350 Recklinghausen

---

The Yugoslav Cracking & distributing Service: We want to swap all over the World. C 64 and PC 128, ober 4000! Programs, also CPLM and 128 Mode. Please send us your List! (Du kannst auch deutsch schreiben) Yu.C.S., Na Produ 38, 6239 1 Prevalse /Yu, Yugoslavia

---

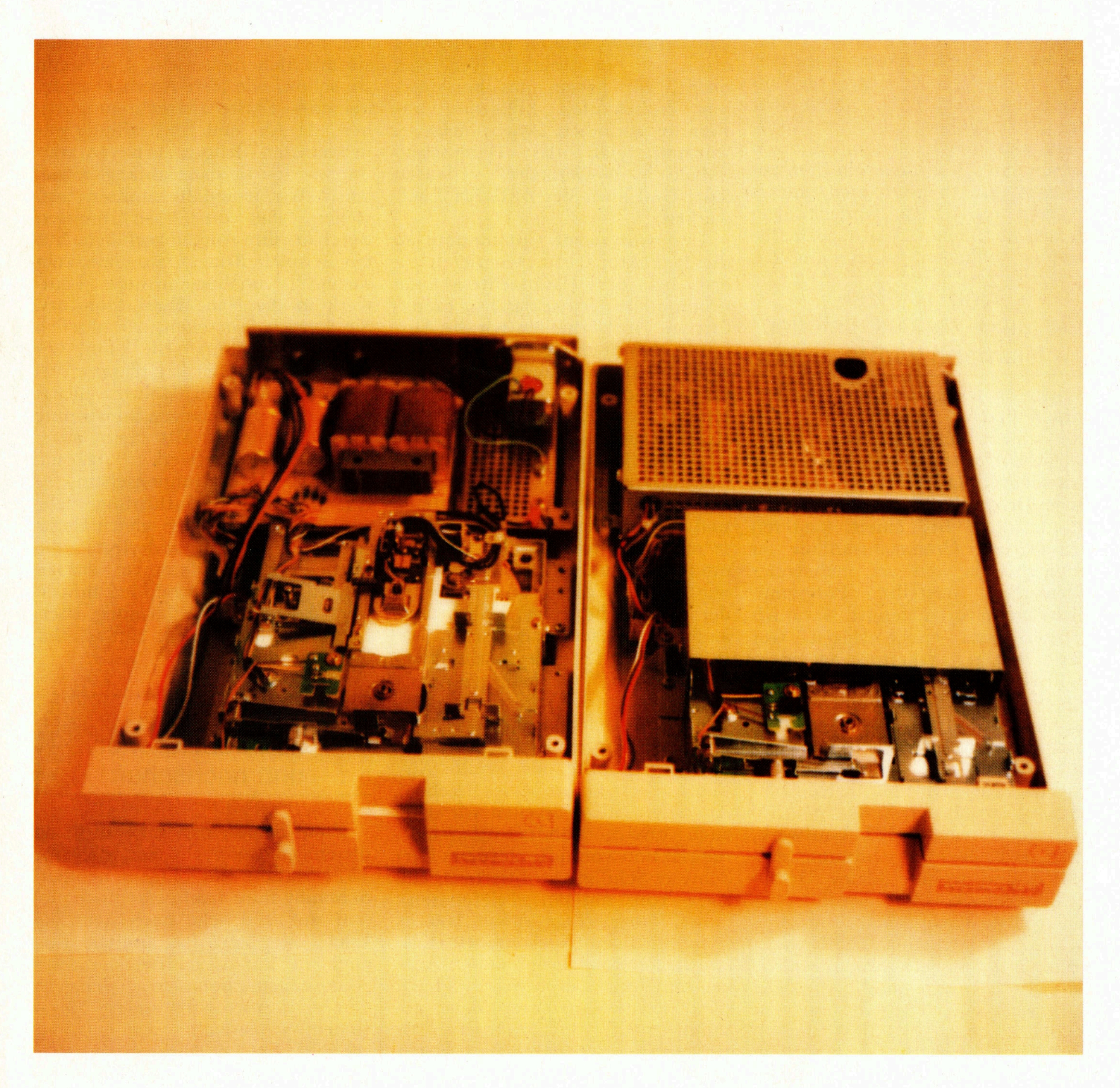
Suche wanted suche
Für VC 20: Datasette (40,— DM); Floppy (250,— DM); 32k-Erw. (100,— DM); grafikfähigen Drucker (300,— DM).
Albert Wirtz, Johannis Str. 2 5180 Eschweiler, Tel: 02403 /65122, Mo., Di., Fr. 19 bis 21 Uhr.

---

Suche zuverl. Tauschpartner (in) für 64/128er-Prg. (Disk.) Suche 64/128er-tauglichen Drucker mit NLQ-Schrift und Interf. bis 800,— DM. M. Dumont, Blankenburgstr. 16, 4242 Rees 3, Tel: 0281/ 200 607 o. 02850/7887

# DIE SELTSAME WANDLUNG EINER FLOPPY

Offensichtlich scheint Commodore wieder in die alte Strategie zu verfallen, gute Geräte anzupreisen und dem Kunden vom Aufbau her völlig andere zu verkaufen. Wie sonst kommt es, daß in „Vorführgeräten", welche der Fachpresse zur Verfügung gestellt werden, Details vorhanden sind, von denen dann der Käufer selbst nichts zu sehen bekommt?



Böse Zungen könnten behaupten, Commodores Fertigungsbetriebe würden weiterhin alte Teile für neue Geräte verwenden. So jedenfalls mutete es uns an, als wir einmal ein 1571 Laufwerk demontierten, welches von uns bei einem Münchener Großmarkt gekauft wurde. Das Erstaunen unsererseits war groß, als uns in dem nagelneuen Laufwerk ein Netzteil entgegenblickte, welches einem altertümlichen Dampfradio alle Ehre gemacht hätte. Als daraufhin (die Neugier wurde immer größer) eine 1571 zerlegt wurde, war dann die Überraschung perfekt. Nichts war zu sehen von Vorkriegsnetzteilen, dafür waren über der Stromeinheit und dem Laufwerk akurate Metallabschirmungen angebracht, von denen in dem anderen Gerät ebenfalls nicht auszumachen war. Diese Abdeckungen dienen dem Schutz vor den für Disketten tödlichen Magnetstrahlen,

## Eine Telefon-Odyssee sonder gleichen — Niemand wußte nichts

welche in Laufwerk und Transformator erzeugt werden. So ist es bei einem abgeschirmten Gerät möglich, im Eifer des Gefechtes mit dem Computer eine beschriebene Diskette auf das Gehäuse zu legen, ohne bei einem späteren Leseversuch Ohnmachtsanfälle zu erleiden, da das in mühevoller Kleinarbeit erstellte Programm nicht mehr vorhanden ist. Natürlich werden jetzt einige fragen, wer denn schon Disketten auf dem Laufwerk lagert, doch das ist nicht die eigentliche Frage. Viel wichtiger erscheint uns doch die Tatsache, daß hier ganz offensichtlich Fehlinformation von Commodore betrieben wird. „Otto-Normalverbraucher" liest in der Zeitschrift XY von dem neuen 1571 Laufwerk, freut sich über Abschirmung desselben und weitere kleine Detailverbesserungen. Also läuft er in das nächste Fachgeschäft und kauft sich dieses Gerät. Zu Hause angekommen probiert er dann natürlich gleich kräftig aus, doch siehe da: Von einer Abschirmung ist nichts zu bemerken, die Versuchsdiskette ist zerstört. Wer würde sich nun an dieser Stelle nicht vera... vorkommen, wenn doch immerhin einige hundert Mark im Vergleich zur 1541 mehr bezahlt werden, nach Aufschrauben und Nachschauen die versprochenen Vorteile jedoch zusammenschmelzen wie Schnee in der Sonne?

Wir kauften uns also NOCH eine 1571, schraubten auf und mußten feststellen, daß auch an diesem Laufwerk der Fortschritt vorbeigelaufen ist. Wenigstens wissen wir jetzt um die eigentliche Bedeutung des Begriffes „Laufwerk". Genau wie bei unserem ersten Kauf fehlte auch hier jede Abschirmung, von einem neuen Netzteil war nirgends etwas zu entdecken. Wer nun meint, wir hätten uns bei unseren Probekäufen Billigware aus Fernost andrehen lassen, dem sei verraten, daß das Typenschild „Made in W.-Germany" verlauten läßt. Und zwar bei BEIDEN Laufwerken.

Und war meint Commodore zu diesen Fakten? Nun, wir riefen zuerst bei der Hauptzentrale von Commodore Deutschland in Frankfurt an. Dort erhielten wir die Auskunft, daß doch eigentlich das Fertigungswerk in Braunschweig hierfür zuständig sei. . Im Werk Braunschweig wurden wir dann an das Auslieferungslager verwiesen, (was ein Auslieferungslager mit Produktionsunterschieden zu tun hat, wurde uns dabei nicht mitgeteilt) und vom Auslieferungslager schickte man uns wieder zurück an Commodore Frankfurt. Dort wandten wir uns nun direkt an den Leiter der Pressestelle, Herrn Hahn, doch auch von diesem war nur zu vernehmen, daß für derlei Dinge „Werk Braunschweig" zuständig sei. Also läuteten wir die nächste telefonische Deutschlandrundreise ein, doch das Ergebnis blieb gleich: Niemand konnte oder wollte Auskunft geben, keiner wußte Bescheid. An dieser Stelle brachen wir dann auch unsere Bundespostbereicherungsaktion vorläufig ab (denn Herr Schwarz-Schilling profitierte als einziger von unseren Telefonaten).

Erst einige Tage später war es dann möglich, (Nachdem ein Kollege nachdrücklich auf diesen Artikel hinwies), daß uns von Herrn Hahn ein kompetenter Mann genannt wurde, welcher unsere Fragen beantworten sollte. Von Herrn Arndt in Braunschweig erhielten wir dann eine Auskunft, welche unsere Verblüffung nur steigern konnte: Das Alt-Netzteil sowie die fehlende Abschirmung seien eine WEITERENTWICKLUNG des 1571er Laufwerkes. Die Abschirmung im „alten" Gerät habe nur deshalb bestanden, da das dort verwendete Schaltnetzteil höhere Abstrahlungswerte erreicht. Die Abschirmung über dem Laufwerk diene lediglich dem Schutz desselben vor Strahlungen von außen, z.B. durch einen Monitor. Was wir aber nun nicht verstanden: Warum konnten diese Abschirmungen denn nicht bestehen bleiben, denn schaden können sie mit Sicherheit nicht. Und wenn sie schon, wie Herr Arndt betonte, keinerlei Existenzberechtigung haben, da Ab- und Einstrahlung nur in kleinsten Mengen auftreten, warum wurden sie dann anfangs eingebaut? Wir jedenfalls mußten feststellen, daß die Abschirmungen durchaus ihre Aufgabe erfüllen. Nur ein Beispiel: Bringen Sie einmal ein „neues", unabgeschirmtes Laufwerk in die Nähe Ihres Monitors. Sie werden feststellen, wie nützlich eine Abschirmung sein kann, wir jedenfalls konnten bei diesem Versuch auf unserem Monitor nichts außer einer erstklassigen Bildstörung mehr erkennen. Mit dem abgeschirmten Laufwerk dagegen herrschte Eitel Sonnenschein, es trat keinerlei Beeinflussung mehr auf. Damit wird auch gleich Herrn Arndts Aussage widerlegt, die „neuen" alten Netzteile würden eine so geringe Abstrahlung aufweisen, daß jede Abschirmung überflüssig sei, bei uns trat nämlich das Gegenteil ein.

## Ran an den Schraubenzieher und nachsehen!

Zum Schluß nun eine Bitte an diejenigen unserer Leser, die eine 1571-Station Ihr Eigen nennen: Schraubt Eure Laufwerke doch einmal auf und seht nach, was bei Euch eingebaut ist. Sollten sich auch in Eurem Laufwerk diverse Änderungen befinden (vielleicht gibt es ja noch ganz andere Mutationen innerhalb der 1671), schreibt uns darüber sowie über ure Erfahrungen mit diesem Gerät.

*Torsten Seibt*

# EIN PROGRAMM MACHT FURORE

**Viele Ehre fand unser Textprogramm für den 128er in Heft 3/86. Hier noch einige Antworten auf immer wieder gestellte Fragen**

Ganz offensichtlich ist unser Textverarbeitungsprogramm bei den Lesern ganz gut angekommen, das beweisen zahlreiche Anrufe bei mir. Dabei ging es zu 99,9 % um den Drucker. Deshalb möchte ich nachträglich noch etwas über die verwendeten Steuersequenzen bemerken.

Bevor ich auf diese Sequenzen eingehe, erst noch einiges „Allgemeines" zum Drucker. Die Grundanforderung an den Drucker ist „Epson kompatibilität", das heißt, daß der Drucker den gleichen Befehlssatz haben muß wie ein Drukker der Marke „EPSON". Dies trifft für viele Drukker zu, aber eben nicht für alle. Eine weitere Forderung ist der Schönschreibmodus, NLQ (Near Letter Quality) genannt. Ebenso müssen alle Schriftarten, die im Programm angegeben sind, vorhanden sein. Dazu gehören auch der Unterstreichungsmodus sowie Super- oder Subscript. Auch über einen „Rückwärtsgang" muß der Drucker verfügen, wenn der Briefkopf in dieser Form gedruckt werden soll. Das in Heft 3/86 abgedruckte Listing ist auf den Drucker „RITEMAN F+" der Fa. Itoh und auf das Interface 92008/G der Fa. Wieseman abgestimmt. Dabei kommt es nicht darauf an, den gleichen Drucker zu verwenden, sondern viel mehr auf das Interface. Denn das Interface muß die Steuersequenzen, richtig übersetzt, an den Drucker senden. Hier ist zwar auch der Befehlsvorrat auch gleich, aber es kann durchaus sein, daß bei anderen Interfaces etwas anderes eingegeben werden muß. So sollten Sie an Hand Ihrer Handbücher die entsprechenden Befehle abändern. Bei dem von mir verwendeten Interface war es notwendig, 3 verschiedene „Kanäle" zum Drukker zu öffnen. Deshalb auch die 3 „OPEN"-Anweisungen in Zeile 80, sie gelten alle nur für den Drucker. Dabei sind eigentlich nur die Sekundäradressen (die jeweils dritte Zahl) wichtig. Über die Sek.adr. 1 können alle Zeichen und Befehle im Drucker angesprochen werden, es können keine Blockgraphiksymbole gedruckt werden.

Sek.adr. 2 gestattet den tatsächlichen Druck von Groß- und Kleinbuchstaben.

Sek.adr. 7 gestattet den Druck von CBM-Blockgraphik. Das heißt also, wenn Sie dem Drukker Befehle übermitteln wollen, so müssen Sie dies (entsprechend dem Listing) über die Kanalnummer 3, Geräteadresse 4 und über die Sekundäradr. 1 tun. Als Befehle sind hier anzusehen: Schriftartwechsel, Unterstreichen, Zeilenvorschub oder Zeilenrücklauf etc. Übergeben werden die Befehle mit „PRINT#3, CHR$ (27); ............" wobei „CHR$ (27)" mit „EC$ " in Zeile 90 definiert wurde. Sie sehen dies in den Zeilen 400 bis 590 mit dem Musterkopf entnehmen. Zeile 480 schaltet den Unterstreichungsmodus ein und Zeile 500 wieder aus. Zeile 550 schaltet „Subscribt" ein und Zeile 570 wieder aus. Die Variable „IN$ " initialisiert den Drucker, da gab es auch einige Probleme, wie ich gehört habe. Der Rest dürfte, so glaube ich, klar sein.

P/B

TEXTVERARBEITUNG 1

```
10 rem textverarbeitung        ====128
20 rem (p) 02.86 cbm + cw team ts =
30 rem ============================
40 rem (c) 11/85                  =
50 rem by peter basch             =
60 rem                            =
70 rem version 7.0  80z/ascii     =
71 rem 128 pc+1541/1570/1571      =
72 rem ============================

80 fast:printchr$(14):be$=chr$(7):p
$=chr$(46):open1,4,2:open3,4,1:open
4,4,7
90 dimz$(660),e$(61):cl$=chr$(147):
cd$=chr$(17):rn$=chr$(18):rf$=chr$(
146):rq$=chr$(145):rb$=chr$(157):sp
$=chr$(32):ec$=chr$(27):in$=chr$(27
)+chr$(64)
100 printcl$rn$" Peter's Text fuer
128'er  < ← > zum  M e n u e.......
....< * > Leerzeile.......":fori=1t
o66:printp$;:next:print"<Zeilenende
":printec$"t"
110 printcl$"..mit Briefkopf  = < B
>  mit Absenderzeile = < A >..oder
Neutral = < N >":print#1,in$;
120 gety$:ify$=""then120
130 ify$="n"then650
140 ify$="b"then280
150 ify$="a"then260
160 goto 120
170 y$="b"
180 printcl$"Schrift ?...< P > Pica
.....< I > Italic.....< K > Komprim
iert.....< S ) Schoen "
190 getb$:ifb$=""then190
```



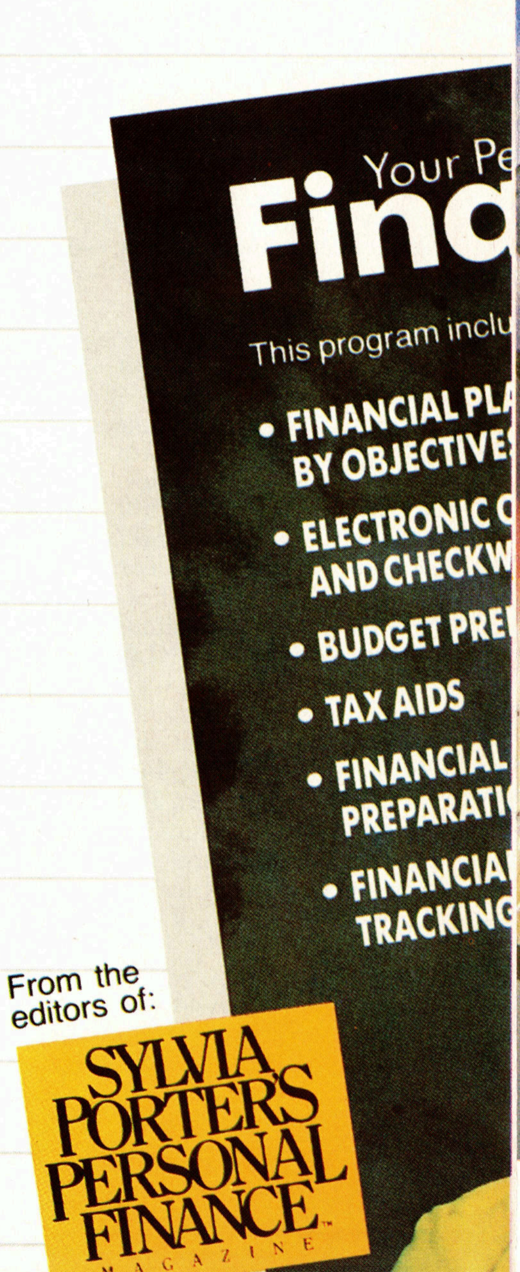

# WOLLEN SIE WISSEN, WO IHR GELD GEBLIEBEN IST?

**Professionelle Software für den 128 PC ist noch sehr rar. Wir entdecken einige gute Anwenderprogramme in den USA. Hier der erste Test.**

Ihr persönlicher Finanzplaner! So könnte man dieses Programm in freier Übersetzung nennen. Denn mit diesem Programm können die Anwender, sofern sie im Besitz eines Commodore 128 plus eines 1571-Laufwerks sind, endlich Ordnung in ihren privaten Finanzhaushalt bringen. Dieses 80-Zeichen-Programm beinhaltet vier aufeinander abgestimmte Vermögensverwaltungsprogramme, die ihnen bei obengenanntem Ziel helfen sollen. Doch sie tun noch mehr. Sie helfen auch bei langfristigen Investitionen, Versicherungen, Renten und sonstigen zukünftigen Geldausgaben. Dabei geht dieses Programm ähnlich einem menschlichen Finanzberater vor. Es benötigt zu Anfang natürlich ausgedehnte Informationen über die finanzielle Lage des „Klienten". Deshalb müssen sie am Anfang den Rechner mit allen Finanzangaben füttern. Dies beinhaltet Dinge wie Prüfen und Sichern ihrer Konten, Rentenpläne, der Aktien und Obligationen, Versicherungsprämien und des Haushaltes. Diese Daten müssen aber nur einmal eingegeben werden, und sind dann durch alle vier Unterprogramme des Finanzplaners zugänglich. Hier gleich mehr zu diesen vier Unterprogrammen, die sich wie folgt auflisten:
- „Transaction Manager"
- „Budget Manager"
- „Asset/Liability Manager"
- „Financial Future"

Dazu ist noch zu sagen, daß auch dieses Programm, wie es bei den meisten heute erwerblichen Programmen üblich ist, durch Über- und Unterprogramme, die in Menüform anwählbar sind, strukturiert ist. Der Transaktionsmanager verfolgt ihre Ausgaben, die sie bar, mit Scheck oder per Kreditkarte tätigen. Er verarbeitet dabei die Informationen von bis zu 5 Bankkonten und 10 verschiedenen Kreditkarten. Weiterhin bietet dieses Unterprogramm folgende Möglichkeiten:
- Eingabe der allgemeinen Informationen
- Hinzufügen der letzten finanziellen Transaktionen. Dazu gehören auch ganz triviale Dinge, wie Einkaufen von Lebensmitteln und Ähnlichem. Der Rechner verlangt dazu allerdings die Eingabe von Codezahlen, um beim Erstellen einer Bilanz die einzelnen Ausgaben einander zuordnen zu können. Diese Zahlen sind im Handbuch äußerst übersichtlich aufgeführt, so, wie man sagen kann, daß dieses Handbuch geradezu ein Vorbild an Übersichtlichkeit ist.
- Analyse der Ausgaben und Einnahmen
- Eingabe automatischer Transaktionen, wie z.B. monatliche Fixausgaben
- Ansehen des Directory einer Diskette
- Suche und Erneuerung von Informationen
- Ansteuerung eines Druckers und Erstellen von Berichten
- Schreiben von Schecks.

Dazu verwendet dieses Unterprogramm eine eigene Datendiskette, die jedoch auch ihre Informationen an die anderen Programme weiter-






geben kann. Diese Diskette kann durch einen eigenen Punkt im Hauptmenü, sowie auch die beiden anderen Datendisketten, initialisiert werden. Das Haushaltsplanmanager-Programm erstellt einen jährlichen Haushaltsplan, wobei es auch das Geld berücksichtigt, das zum Erreichen von Langzeitzielen benötigt wird, wie zum Beispiel die Ausbildung von Kindern, ein neues Auto oder eine größere Urlaubsreise. Auch kann man mit diesem Unterprogramm sich Balkendiagramme zur Veranschaulichung der finanziellen Situation erstellen. Auch dazu muß man eine eigene Datendiskette erstellen. Unterprogramm drei, der Aktiva/Passiva-Manager, schlüsselt Ihnen Ihre Aktiva und Passiva-Posten Ihrer Haushaltsbilanz auf und sagt Ihnen, wieviel Geld Sie noch besitzen, oder ob Sie schon alles ausgegeben haben. Auch hier kann man sich wieder Ausdrucke der aktuellen Bilanzen erstellen lassen. Dazu ist wieder eine eigene Datendiskette nötig. Das vierte Unterprogramm behandelt die finanzielle Zukunft. Dies ist sicherlich das interessanteste Programm im Finanzplaner, da man sich hier selbst Ziele und deren Prioritäten setzen kann. Hier kann man zukünftige größere Anschaffungen eingeben, wann diese getätigt werden sollen und welchen Betrag sie kosten werden. Das können aber auch Punkte wie eine vorgezogene Rente oder Ähnliches sein. Das Programm errechnet dann, wieviel Geld sie dafür jährlich auf die Seite legen müssen und berücksichtigt dabei ihre aktuelle finanzielle Situation genauso wie die Wertminderung des Geldes. Es sagt ihnen aber auch, wenn das gesteckte Ziel in dem angegebenen Zeitraum nicht zu realisieren ist. Das Programm zeichnet sich durch eine ausgesprochen benutzerfreundliche Bedienung aus. Es verwendet sehr stark die sogenannte Fenstertechnik als Anzeige auf dem Bildschirm. Einzelne Funktionen werden mit der Cursorsteuerung angefahren und mittels der Returntaste aufgerufen. Zum Aussteigen und der Rückkehr ins Hauptmenü genügt ein Drücken der ESC-Taste. Auch ist es möglich, eine Hardkopie des jeweiligen Bildschirmes mit dem Drucker zu erstellen.

Eine weiter gute Idee der Programmierer ist es, einen Taschenrechner im Programm zu installieren. Dieser kann jederzeit durch Drücken der F7-Taste auf den Bildschirm gebracht werden und wird dann ganz normal mit der Tastatur bedient. Ein weiterer Druck auf die ESC-Taste und der Rechner verschwindet wieder. Mittels einer eigenen „Memo Pad" Funktion ist es sogar möglich, kurze Informationen zusätzlich abzuspeichern. Sehr lobenswert ist auch, daß vor der Initialisierung der Datendisketten dem Benutzer angegeben wird, wenn er eine bereits beschriebene Diskette in das Laufwerk gelegt hat und ihm wird nahegelegt, daß er erst deren Inhalt überprüfen soll, bevor der Rechner diese neu initialisiert. Dies sind nur einige der sehr nützlichen und durchdachten Zusatzmöglichkeiten zu den vier Hauptunterprogrammen, die dieses Programm bietet. Dazu kommen noch die sehr einfache Bedienbarkeit des Programmes und die einfache Lesbarkeit des übersichtlichen Handbuches, das trotz seines Umfanges (auch auf Grund eines ausführlichen Beispieles) auch von Nichtfinanzexperten gelesen werden kann.

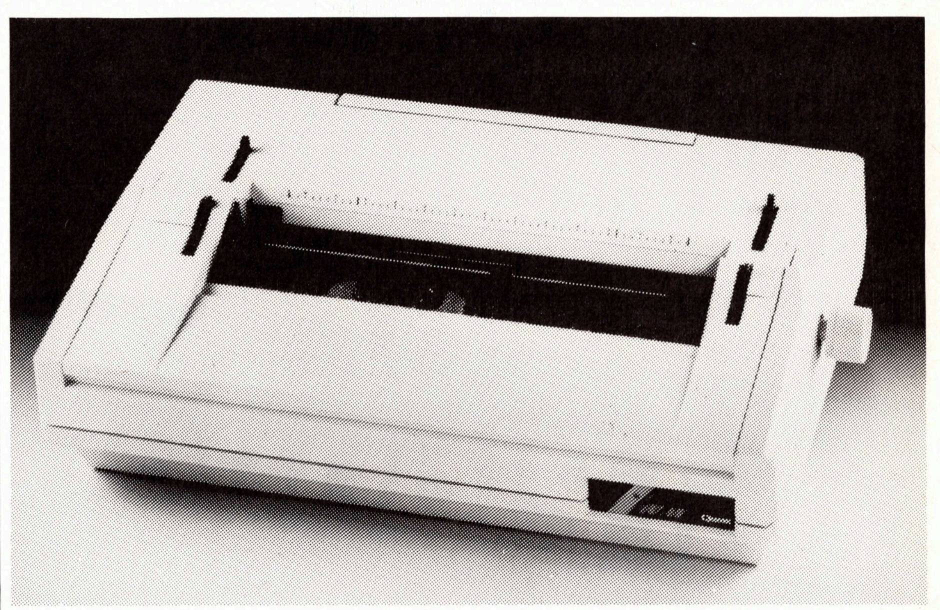
**Financial Planner**

**Hersteller:**
**Timeworks**
**444 Lake Cook Road**
**Deerfield, Il. 60015**
**USA**

**Preis:**
**59.95 US-Dollar**

Dazu bietet der Vertreiber noch einen besonderen Service: Eine kostenlose Telefonnummer, an die sich der Anwender in Fragefällen wenden kann. Alles in allem kann man sagen, daß dieses Programm eine sehr gelungene Version eines privaten Finanzhaushaltsprogrammes darstellt, das man guten Gewissens weiterempfehlen kann. Programmqualität, die man sonst nur von PC-Programmen gewöhnt ist. Doch zum Schluß noch ein kleiner Wermutstropfen: Das Programm ist bisher leider nur auf dem amerikanischen Markt (in engl. Sprache) erhältlich.

# SEITE UM SEITE FLOTT SERVIERT



Der Aufbau des „Easyfeed 230" geht relativ problemlos vonstatten. Es besteht keine elektronische Verbindung zum Drucker. Lediglich ein Adapter-Gerüst wird in die Druckermechanik eingehängt. Dabei greift ein Zahnrad in das an der Walze befindliche Zahnrad hinein. Auf den Adapter, der für mehr als 80 Druckertypen lieferbar ist, wird dann der eigentliche Papiereinzug gehängt, der ebenfalls mit einem Kunststoffzahnrad ausgerüstet ist. Damit sind bereits alle notwendigen Verbindungen hergestellt. Die Papierzuführung wird über die Endlosformularsteuerung des Druckers gesteuert.

Wer viel zu drucken hat und das nicht nur auf Endlospapier erledigen will, wird irgendwann mit dem Gedanken spielen, sich einen Einzelblatteinzug anzuschaffen. Einige Firmen haben sich auf derartiges Zubehör wie Lowcost-Drukker spezialisiert. Wir haben einen Einzug der Schweizer Firma Multimatic mit dem Star SR-15 Matrixdrucker getestet.

Mit einem kleinen Trick wird der automatische Druckerstop beim Öffnen der Abdeckhaube überlistet: Ein Magnet, auf den Magnetschalter geklebt, simuliert „Haube geschlossen". Kleiner Nachteil: der Staub- und Geräuschschutz läßt sich nun nicht mehr ganz schließen, so daß der Drucker etwas lauter wird.

## Saubere Arbeit

Rund 100 Blatt Papier lassen sich unter die beiden Gummirollen klemmen und werden auch präzise eingezogen. Selbst während des laufenden Drucks kann Papier nachgefüllt werden. Schwierigkeiten machte zunächst bei mehrseitigen Texten der Übergang von einer zur anderen Seite. Schuld daran war aber nicht das Gerät, sondern die Anleitung, die über die notwendigen Druckereinstellungen (Endlospapiermodus) kein Wort verliert. Das als „unglaublich einfach" gepriesene Verfahren des „top of form" (etwa: Fixierung des Seitenanfangs) bestand den Praxistest mit Glanz und Gloria.

Das verwendete Programm, in unserem Falle eine Textverarbeitung, muß lediglich auf die richtige Seitenlänge eingestellt werden. Dann beginnt der Drucker bei jedem Blatt exakt an der gleichen Stelle seine Arbeit. Doppelblatteinzüge haben wir bei dem verwendeten 80 g-Papier nicht festgestellt. Kleiner Schönheitsfehler in der ansonsten gut nachzuvollziehenden Anleitung: Der für das Papierende zuständige DIP-Schalter ist falsch bezeichnet (C-1 statt B-1).

*Bezugsquelle: Multimatic Vertriebsgesellschaft, Postfach 22, D-7214 Zimmern o.R., Tel.: 0741-22132.*

Preis des tadellos funktionierenden Einzelblatteinzugs: runde 850,– DM. Bei Druckerwechsel braucht nur ein neuer Adapter (ab 90,– DM) zugekauft werden. Das sind zweifelsohne angesichts des geringen Materialaufwandes und der vergleichsweise einfachen Mechanik recht stolze Preise. Aber bei Profidruckern muß man allein für einen Einzelblatteinzug noch mehr hinblättern, als für den kompletten Star-SR 15 – runde 2000,– DM.

# VERDIENEN SIE GELD MIT IHREM COMPUTER!

Haben Sie einen Commodore VC 20 oder C 64? Einen 16/116/Plus 4? Oder einen 128? Können Sie programmieren? In Basic oder Maschinensprache? Dann bietet COMMODORE-WELT Ihnen die Möglichkeit, mit diesem Hobby Geld zu verdienen!

**Wie? Ganz einfach. Sie senden uns die Programme, die Sie für einen Abdruck als geeignet halten, zusammen mit einer Kurzbeschreibung, aus der auch die verwendete Hardware – eventuelle Erweiterungen – benutzte Peripherie – hervorgehen muß (Schauen Sie sich dazu den Kopf unserer Programmlistings an.)**

**Benötigt werden: Zwei Listings des Programms sowie eine Datenkassette oder Diskette! Wenn die Redaktion sich überzeugt hat, daß dieses Programm läuft und sich zum Abdruck eignet, zahlen wir Ihnen pro Programm je nach Umfang bis zu DM 300,–!**

**Sollten Sie keinen Drucker haben, genügt der Datenträger.**

Sie erhalten Ihre Kassette/Diskette selbstverständlich zurück, wenn Sie einen ausreichend frankierten Rückumschlag mit Ihrer Adresse beifügen.

**Bei der Einsendung müssen Sie mit Ihrer Unterschrift garantieren, daß Sie der alleinige Inhaber der Urheber-Rechte sind! Benutzten Sie bitte anhängendes Formular! (Wir weisen darauf hin, daß auch die Redaktion amerikanische und englische Fachzeitschriften liest und „umgestaltete" Programme ziemlich schnell erkennt).**

**Um Ihnen die Arbeit zu erleichtern, finden Sie hier ein Formular. Sie können es ausschneiden oder fotokopieren.**

## PROGRAMMANGEBOT

Name des Einsenders: ______________________

Straße/Hausnr./Tel.: ______________________

Plz/Ort: ______________________

Hiermit biete ich Ihnen zum Abdruck folgende(s) Programm(e) an:

______________________

______________________

______________________

______________________

Benötigte Geräte: ______________________

______________________

Beigefügt ( ) Listings ( ) Kassette ( ) Diskette

**Ich versichere, der alleinige Urheber des Programmes zu sein!**

Hiermit ermächtige ich die Redaktion, dieses Programm abzudrucken und wirtschaftlich zu verwerten. Sollte es in den Kassetten-Service aufgenommen werden, erhalte ich auch dafür eine entsprechende Vergütung.

______________________

Rechtsverbindliche Unterschrift

**CW/CBM** Postfach 1107
8044 Lohhof